Mechanical Computing in Microelectromechanical Systems (MEMS)

THESIS

Kenneth C. Bradley
Second Lieutenant, USAF

AFIT/GE/ENG/03-04

DEPARTMENT OF THE AIR FORCE

AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the United States Air Force, the United States Department of Defense or the United States Government.

AFIT/GE/ENG/03-04

Mechanical Computing in Microelectromechanical Systems (MEMS)

THESIS

Presented to the Faculty of the Graduate School of Engineering and Management
of the Air Force Institute of Technology
Air University
In Partial Fulfillment of the
Requirements for the Degree of
Master of Science in Electrical Engineering

Kenneth C. Bradley, B.S.E.E.
Second Lieutenant, USAF

25 March 2003

Approved for public release; distribution unlimited.

AFIT/GE/ENG/03-04

Mechanical Computing in Microelectromechanical Systems (MEMS)

THESIS

Kenneth C. Bradley, B.S.E.E.

Second Lieutenant, USAF

Approved:

13 MAR 03

Capt Paul E. Kladitis, Ph. D. — Date
Assistant Professor of Electrical Engineering
Thesis Advisor

James A Lott — 13 MAR 03

Lt Col James A. Lott, Ph. D. — Date
Professor of Electrical Engineering
Committee Member

Henry B. Potoczny — March 13, 2003

Henry Potoczny, Ph. D. — Date
Professor of Computer Science
Committee Member

Acknowledgements

I would like to thank my sponsors (Dr. Scott Roberson of AFRL/MNMF, Dr. Robert Ewing of AFRL/IFTA, and the AFIT Cooperative Research and Development Agreement Funds, Emerging Air Force Thrusts Proposal ENR Proposal # 2002-027) for their assistance with this research. Without their support and funding, this thesis would not have been possible. I would also like to recognize the direction of my advisor, Capt. Paul Kladitis, and the support and guidance that I received from my committee members, Lt Col James Lott and Dr. Henry Potoczny. I would also like to thank my wife and our son for putting up with my absence during a good portion of this past year. Many other individuals have helped me with my research. Rick Patton and Bill Trop were always willing to help out in the clean room and test lab, Dr. Baker was a tremendous help when I was attempting to model my devices mathematically, Dr. Vi at AFRL/SND helped with taking SEMs, and Charlie Powers coordinated purchasing of equipment for my research. I would like to also thank the microsystems group for being there throughout the process (Jared, Joel, Gabe, and Ron). We got through this together.

Table of Contents

Page

List of Figures

Figure Page

Figure Page

List of Tables

AFIT/GE/ENG/03-04

Abstract

Mechanical computing devices in polysilicon-based microelectromechanical systems (MEMS) were designed with the goal of developing computing devices for harsh environments, such as those with high dose radiation and high temperatures, as well as devices that may be able to interface with molecular or biological computer systems. The devices that were designed include both analog and digital computing devices. The analog devices include integrators, differentials (summers), multipliers, and those that perform trigonometric functions. The digital devices that were designed are inverters, NAND, NOR, and XOR logic gates. Analog-to-digital (A-to-D) and digital-to-analog (D-to-A) converters were also designed. The designs were submitted to a commercial surface micromachining foundry to be fabricated. The completed MEMS devices were then released and tested to determine proper operation. Of the mechanical devices that have been fabricated and tested, a functioning inverter, sine function device, cosine function device, and digital-to-analog converter have been demonstrated.

Mechanical Computing in Microelectromechanical Systems (MEMS)

I. Introduction

Mechanical computing is not a new concept, but the ability to fabricate moving mechanical structures using the microelectromechanical systems (MEMS) technology has opened the door for mechanical computing at a much smaller scale than that employed by the mechanical devices of the past. This thesis consists of an investigation of purely mechanical analog and digital computing devices in MEMS. By way of introduction, this chapter presents the motivation behind this work, gives a brief background on previous work in the area of mechanical computing in MEMS, lists potential applications of purely mechanical computing devices, details the scope and approach of this research, summarizes the accomplishments and results, and gives an overview of the thesis as a whole.

1.1 Motivation

The benefits of purely mechanical computing in MEMS has given rise to the motivation for this research. This motivation is largely influenced by the desire to have computers that are able to operate in harsh environments, where high temperature or radiation is present. This is not the only motivating force behind this research, however. Other motivations are the need for smaller computer systems, and computing devices that can be used as an interface between biological or molecular systems and larger mechanical or electrical systems. This section will briefly discuss each of these motivations, beginning with a discussion on harsh environments.

Harsh environments are commonly defined as environments in which microelectronic devices are prone to failure. As it relates to this thesis, harsh environments will specifically refer to high temperature or radiation environments. Current microelectronic computing devices are fabricated in silicon (Si). Due to its relatively small bandgap, these devices have reduced performance above 200 °C [1]. Consequently, research in the area of wide bandgap semiconductor materials, such as silicon carbide (SiC), is being undertaken in an effort to increase the reliable operating temperature of the microelectronic computing systems in use today. Due to the adverse effects of radiation, an expensive radiation hardening process is used to protect microelectronic computing devices that are to be used in radiation environments. It is proposed that purely mechanical computing devices may be suitable for computation in both high temperature and radiation environments using the presently available silicon (Si) technology and without the added expense of radiation hardening. These devices could be fabricated as microelectromechanical systems (MEMS) using a commercial fabrication process, and would have mechanical inputs and outputs.

The MEMS mechanical computing devices are passive devices that do not require wires or power supplies to operate. They are actuated by external mechanical inputs, and supply a mechanical output. Schematically, this purely mechanical computing paradigm can be represented as shown in Figure 1.1, where the mechanical "signal" transitions from input to output in one step [2]. This results in a more consistent device architecture as compared with that employed by current microelectronic computing devices, which are active devices that require a rather bulky power supply and signal conditioners at both input and output. Figure 1.2 schematically portrays the current microelectronic computing paradigm. It is proposed that purely mechanical computing devices can be made intrinsically smaller than their

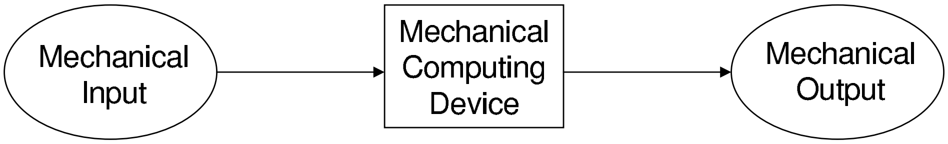


Figure 1.1 Purely Mechanical Computing Paradigm.

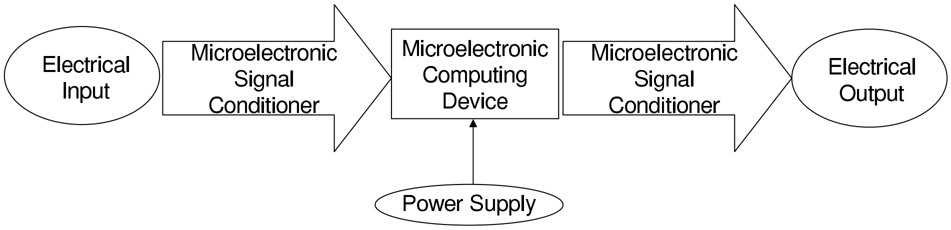

Electrical Input
Microelectronic Signal Conditioner
Microelectronic Computing Device
Microelectronic Signal Conditioner
Electrical Output
Power Supply

Figure 1.2 Microelectronic Computing Paradigm.

microelectronic equivalents due to the paradigm shift that this transition would entail.

This sensor-to-actuator size reduction can be thought of as an intermediate step towards achieving the goals of nanotechnologists and molecular scientists; that is, computing at the molecular scale. In fact, another benefit to purely mechanical computing in MEMS is that it may be a perfect interface between molecular or biological systems and devices at the larger macro-scale. Specific applications of this research will be discussed more fully in Section 1.4, as the purpose of this section is to present the fundamental motivation for this research.

With a clear understanding of the motivation for purely mechanical computing in MEMS, it would be beneficial at this point to examine the current status of the research in this area. Section 1.2 will give a brief background of mechanical computing in microelectromechanical systems (MEMS).

1.2 Brief Background of Mechanical Computing in Microelectromechanical Systems (MEMS)

The first attempts at mechanical computing using microelectromechanical systems (MEMS) components were not purely mechanical. They were used to mechanically convert electrical inputs to mechanical outputs [3–10]. Each of these is discussed briefly here, with a more complete analysis of the work contained in Section 2.9.

Toshiyoshi's microelectromechanical digital-to-analog converter (MEMDAC) used a series of electrical inputs in the form of four digital voltages to produce a mechanical displacement of a series of electrostatic actuators to give a mechanical analog output [3–5]. Yeh's mechanical digital-to-analog converter performed a similar operation with the use of thermal actuators instead of electrostatic actuators [6]. Hirata's micromechanical switch used electrostatic forces to actuate a mechanical switch for logic operation, enabling logic operations to be performed with only one switch [7, 8]. Kruglick used MEMS relays with arrays of electrothermal actuators to perform the operation of a field-effect transistor (FET) [9, 10]. The operation of both n- and p-FETs were realized with the same relay structure.

Although each of these devices involved mechanical computing using MEMS components, the first publication in the MEMS field on purely mechanical computing, that is, mechanical inputs resulting in mechanical outputs, was Kladitis' design of micro-mechanical logic gates [2]. This thesis follows Kladitis' work in an effort to address an underlying problem with microelectronic computing.

1.3 Problem Statement

One of the main problems with microelectronic computing is the tendency for devices to fail in harsh environments. As was discussed in Section 1.1, purely mechanical computing in MEMS may be the solution to this problem. Another problem that MEMS mechanical computing devices may be able to solve is the need to interface with biological or molecular computers. This thesis serves as the next step towards achieving purely mechanical computing devices in MEMS, which may be used in future real-world applications. The next section details some of these potential applications.

1.4 Applications

The application of this research to real-world problems is not specific to any particular group. At the present time, nuclear and space applications are the most likely beneficiaries of this work. However, it is envisioned that applications which meet any or all of the following criteria may benefit from this research, as well:

(i) High temperature operating environment

(ii) Radiation environment

(iii) Miniaturization of a sensor-to-actuator computing system

(iv) Interface between macro- and molecular or biological systems.

For space applications, such as deploying a satellite into orbit, the first three criteria come into play. First, extreme temperatures are encountered in the space environment. Second, many forms of radiation are present in space. Third, reducing the size of the computing components reduces the mass of the satellite, and the amount of fuel needed. Essentially, the purely mechanical computing devices would not only function in the harsh space environments, but would do so at a reduced cost.

For nuclear applications, the second criteria is definitely met, and the first is a real possibility. For example, a safe-and-arming device for a nuclear weapon may require some computation be performed to determine whether it should detonate. A MEMS mechanical computing device would not only be able to withstand the radiation, but also the high temperature that would result in the process of launching the weapon.

Nanotechnologists are currently working on molecular computing from two directions. These are the top-down and bottom-up approaches to fabrication of nanostructures. The top-down approach takes advantage of current microelectronic processing and is focused on improving minimum device dimension. The second approach begins with the molecules themselves, and attempts to build devices molecule by molecule. Due to the difficulties associated with the top-down approach, the bottom-up approach appears to be more likely to lead to molecular computing. However, building structures one molecule at a time will not lead to an efficient fabrication process for computing devices. Consequently, self-assembly is required. A MEMS computing device could be used to interface with the solution of molecules, allowing the molecules to form in a more ordered manner.

Similarly, MEMS computing devices could be used to interface between the macroscopic world in which we live and a biological system. Consider, for example, the powerful memory capabilities of DNA. Significant amounts of data could be stored in a biological DNA computer, but in order for these data to be accessed by the macro-computers that we are accustomed to, an interface would be needed. A MEMS computing device could very well be used as the interface to access the memory stored in the DNA computer system.

As can be ascertained from these examples, purely mechanical MEMS computing devices can be used in numerous applications. At the present time, some of these applications may seem to more closely resemble science fiction than science fact, but such is the case with state-of-the-art research. This thesis could not possibly address every application of MEMS mechanical computing devices, just as it could not possibly result in the development of every possible computing device. As such, the next section presents the scope and approach for this research.

1.5 Thesis Scope and Approach

In an effort to contribute to the development of purely mechanical computing devices in MEMS, the scope of this thesis was to design both analog and digital computing devices that could be fabricated using a commercially available fabrication process.

The devices were designed using L-Edit [11], a mechanical design software layout tool, and the design layouts were sent to the MUMPs® [12] foundry for fabrication. The MEMS devices were then released and tested to verify correct operation.

This research has resulted in the design of purely mechanical digital and analog computing devices on the same size as present microelectronic devices. It is the hope of the author that this foundational research will lead to further developments in the area of mechanical computing in MEMS, which may lead to the replacement of microelectronic computing devices with MEMS mechanical computing devices for high-temperature or radiation-sensitive applications in the future. A further hope is that this research will lead to the miniaturization of mechanical computing devices at the nano- and molecular scales.

1.6 Accomplishments

A toolbox of both analog and digital mechanical computing devices has been designed in microelectromechanical systems (MEMS). Analog mechanical computing devices include differentials (adders), multipliers, integrators, and devices that perform trigonometric functions. Digital mechanical computing devices include inverters, NAND, NOR, and XOR logic gates. Digital-to-analog (D-to-A) and analog-to-digital (A-to-D) converters have also been designed. All devices have purely mechanical inputs and outputs. For the devices that have been fabricated, testing has been performed to determine correct operation. A functioning inverter, sine function device, cosine function device, and digital-to-analog converter have been demonstrated. Designs of all other devices have been submitted for fabrication and will be tested in future work.

1.7 Uniqueness of Thesis

The idea of purely mechanical computation in MEMS was first published by Kladitis in reference to logic gates [2]. However, this research is unique in that it does not restrict itself to logic operations. Purely mechanical analog computing devices have been designed, fabricated, and tested, as well. The next section presents an overview of this thesis.

1.8 Thesis Overview

This thesis is organized into seven chapters. This introduction serves as the first chapter, which includes the statement of the problem that is being addressed as well as a brief overview of research in MEMS that has led to this effort.

The second chapter provides a more detailed background of the topic, including a historical overview of the history of mechanical computing, a presentation of related work in the area of nanotechnology and molecular computing, an analysis of high temperature and radiation effects on microelectronic devices, a discussion of three MEMS fabrication processes, an in-depth explanation of the MUMPs® foundry processing steps, and a detailed summary of publications relevant to mechanical computing in MEMS.

The third chapter consists of the author's designs that were made using L-Edit and submitted to Cronos for fabrication. A brief background on each of the designs with emphasis on the origin of each device's inception is also included.

The fourth chapter consists of mechanical modelling of the logic gates and their elemental components in an effort to determine potential switching speeds. Element models have been derived and utilized to model entire devices as systems of elements.

The fifth chapter provides details of the experimental procedures that were followed in performing this research, including the processing steps that were followed to release the MEMS devices, and the experimental setup that was used to perform the operational testing.

The sixth chapter gives the results of the experimentation, including an analysis of the designs that were fabricated and tested. This analysis specifically addresses the operation of the devices, and discusses problems found and potential solutions.

The seventh chapter serves as the conclusion of this thesis. It includes an analysis of the successes and failures that were encountered during the course of this research, as well as recommendations for future work related to this topic.

II. Background

The purpose of this chapter is to provide a thorough background on mechanical computing. This background includes the history of mechanical computing and an in-depth look at recent work in MEMS on this topic. It also covers topics associated with potential applications for this work, such as harsh environments and nanotechnology. Furthermore, general information is provided regarding the three classes of MEMS fabrication processes, as well as specific details of the fabrication process used for this research. As with any endeavor, one must first examine the history before true progress can be made for the future.

2.1 History of Mechanical Computing

The history of computing began long before the transistor and electronic devices were used to do computation. These early computers were purely mechanical, and they enabled people to perform tedious tasks and complex computations.

Around 1000 B.C., the earliest version of what is known today as the abacus was invented [13]. The abacus (see Figure 2.1) consists of a series of rods with balls, with the position of each ball representing a number. The balls are slid to perform the basic arithmetic operations of addition, subtraction, multiplication, and division.

In Peru, the Incans used a different version of mechanical computer to perform numerical calculations [13]. They invented the quipu in 1400 A.D., which consisted of a series of knotted cords. One common use of the quipu was in census taking.

A few hundred years later, in 1617, John Napier (1550-1617), a Scottish scholar, came up with a multiplication device, known as Napier's bones (see Figure 2.2), which consisted of sliding rods in the form of a multiplication table [15]. His work in the area of logarithms also led to William Oughtred's invention of the slide rule (see Figure 2.3) in 1621.

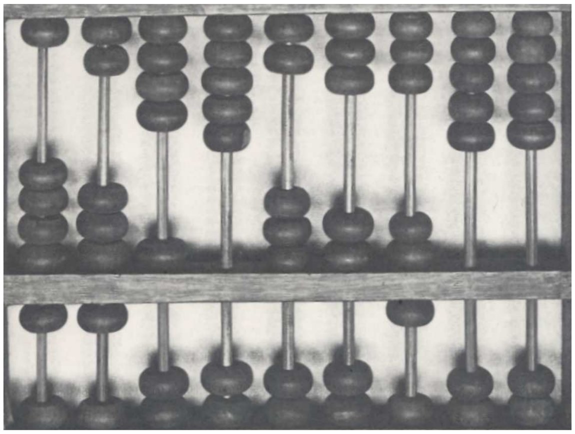


Figure 2.1 Abacus [14].

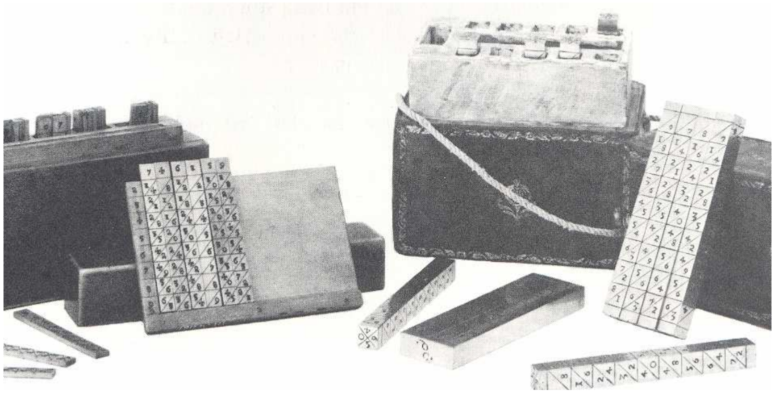


Figure 2.2 Napier's bones (1617) [14].

In 1642, Blaise Pascal (1623-1662) became frustrated with the tedious nature of counting coins in his job as a tax clerk [13]. This resulted in the invention of Pascal's Calculator, which was a mechanical adder for counting money by simply pushing buttons representing the coin that was to be counted. Within the next few years, Pascal invented the Pascaline (see Figure 2.4), the first mechanical calculator [16].

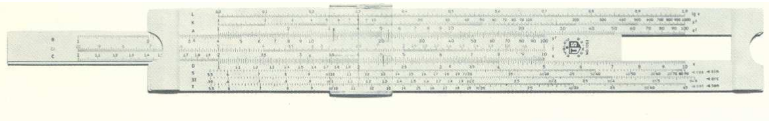


Figure 2.3 Slide rule [15].

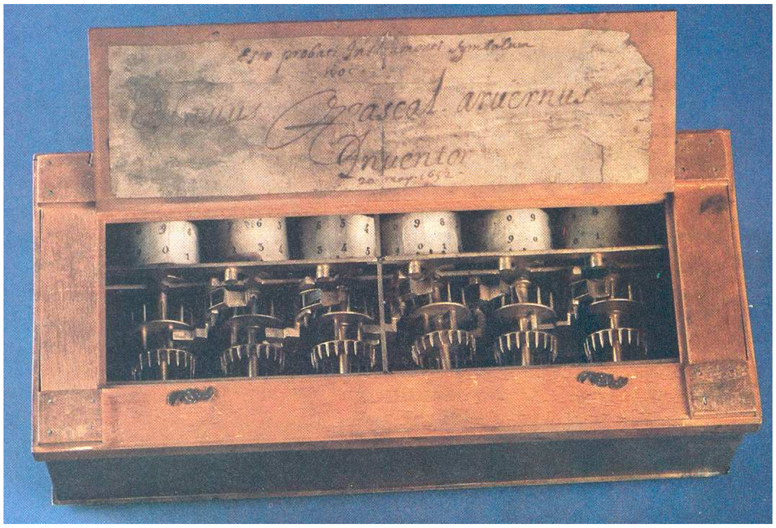


Figure 2.4 Pascaline [14].

Joseph Jacquard (1752-1834) invented the Punchcard Loom in 1801 for the weaving industry [13]. It operated with the use of punchcards that positioned the thread for weaving.

In 1822, Charles Babbage (1792-1871), an English mathematician, proposed his idea of the Difference Engine, which would “solve polynomial equations by calculating successive differences between sets of numbers” [15]. Babbage’s Difference Engine was “designed to calculate with numbers, to store information, to select different ways of solving problems according to the most efficient approaches, and to deliver printed solutions to problems both during the solutions and at their conclusion” [13]. The Difference Engine consisted of a “mill”, which performed the calculation, a receiver, a printer, a “store” to hold the information, and a device to transfer the information between components [13]. Babbage’s other notable invention was the Analytic Engine (see Figure 2.5), which was a decimal-based general purpose mechanical computer [17]. He began to design the Analytic Engine in 1834, with the use of punchcards for the two different types of inputs, information and instructions.

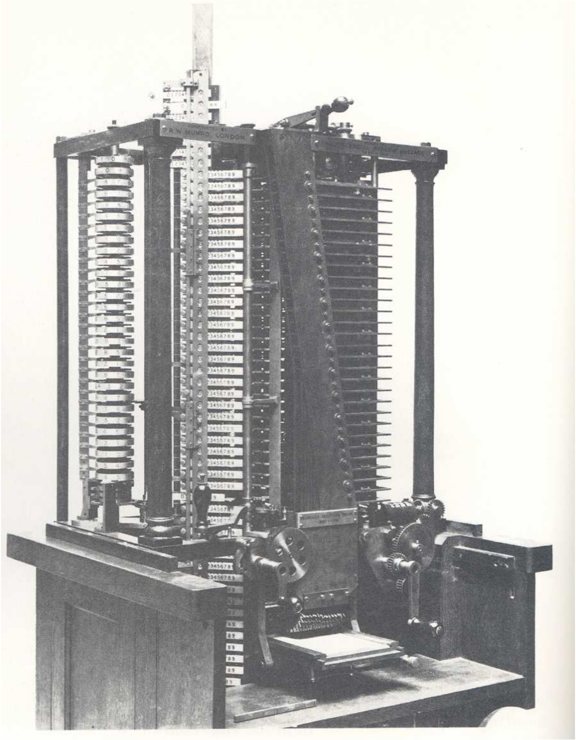


Figure 2.5 Charles Babbage's Analytic Engine [14].

After the U.S. Census was taken in 1880, it took seven years to compile the data, since it was done manually. The increasing population and an interest in gathering more data, such as race, employment, and literacy, suggested that by the 1890 census the data processing would take too long to complete. The Census Office held a competition to try to find a better method [17]. Herman Hollerith, in winning the competition, used punchcards (see Figure 2.6) to store the census data.

The needs of the United States Navy played a major role in the development of mechanical computing in the first half of the twentieth century. In 1935, Arma Corporation developed a torpedo mechanical computer, which was to be used on submarines to "establish [the torpedo's] course, speed, and depth. The torpedo itself contained several small mechanical analog computers. These computers included the following mechanical devices: 1. The course control system that activated a rudder. 2. A computer to determine the course angle for collision with the target. 3. A

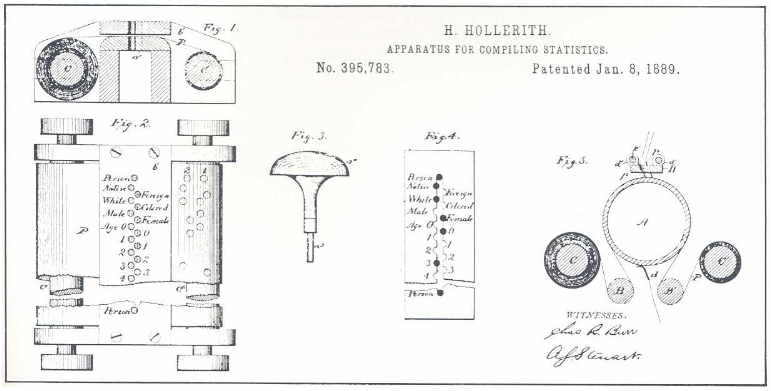


Figure 2.6 Herman Hollerith's statistic machine for Census taking [14].

depth-control system, relying on a diaphragm to measure depth (water pressure) and a pendulum to measure rate of change of depth" [18].

Mechanical bombsight computers, such as the Ford Bombsight mechanical computer in Figure 2.7, were developed for bombers at the end of World War I [18]. The bombsights were small and precise. Similarly, sights for guns on Navy ships were

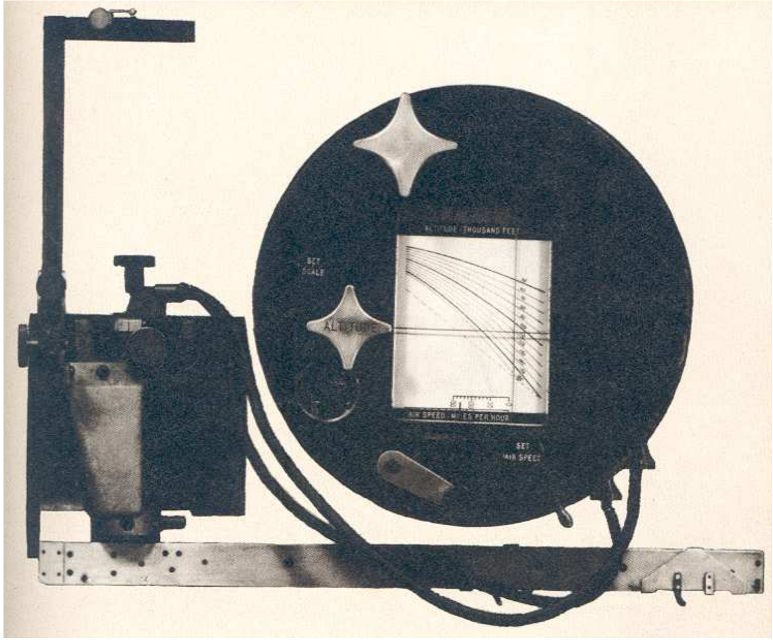


Figure 2.7 Ford Bombsight mechanical computer [17].

designed as well. Other mechanical computers that were built during this time period include Ford Instrument Company's mechanical analog flight simulator (1945), Arma's inertial guidance system for the Atlas missile, and an analog mechanical computer that determined the range of the German V2 rocket [18]. Although mechanical computing devices have a very pronounced role in the history of computing, the majority of mechanical computing devices were replaced by faster electrical computers by the end of World War II. Despite this fact, mechanical computing "has not, even yet, died out" [18].

With the emergence of the transistor and the integrated circuit technology, electrical computing devices have become the leading area of research and development for the computing industry. Active research in the area of mechanical computing has virtually stopped.

Given this fact, one might question the need to discuss the future of mechanical computing. However, there are many areas in which mechanical computing may have a bright future. Among these are nanocomputing, which includes both molecular and biological systems, and computing in harsh environments, which encompasses high temperature and radiation. The next few sections will present details on these topics.

2.2 Nanocomputing

In the area of nanocomputing, Richard Feynman and K. Eric Drexler have written on the topic of computing at the submicroscopic scale. Drexler has focused much of his discussion in the area of nanocomputing with rod logic. Rod logic behaves similar to transistor (CMOS) logic in the sense that it performs the binary logic operations, but mechanical displacements represent the binary values of 1 and 0 rather than electrical voltages. Drexler has shown, theoretically, that nanomechanical computing can be performed using rod logic with switching times on the order of 0.1 ns and with energy dissipation much less than kT per gate at room temperature [19].

Feynman's focus on nanocomputing has been on the idea of reversible gates. The binary logic that is used in transistors today is irreversible. From the output, the inputs cannot be determined. For example, the output of an AND gate is one if both of its inputs are one; otherwise the output is zero. Assuming the output is known to be zero, the inputs are indeterminable. Input A could be one and B could be zero, B could be one and A could be zero, or both A and B could be zero. The problem with this irreversibility is that in order to compute on an atomic scale, the laws of quantum mechanics must be obeyed. One of these laws is that "atomic physics is reversible" [20]. Therefore, transistor logic is not a valid option for computing at the atomic scale. Feynman's solution to this problem is to have logic gates with three inputs and three outputs (see Figure 2.8). Two of the inputs are A and B, as before. The outputs correspond to the inputs, A and B, and the resulting output to the logic gate is C′.

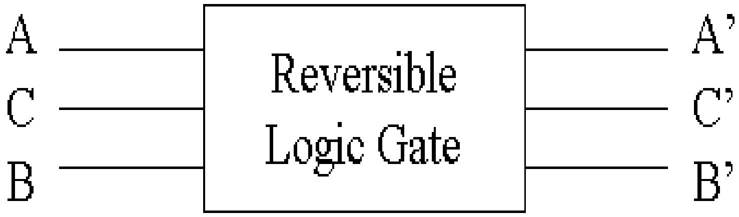


Figure 2.8 Reversible Logic Gate.

Efforts have been made with some success in the area of molecular and biological computing, but many challenges have been discovered along the way. Both of these topics will now be discussed, beginning with molecular computing.

2.3 Molecular Computing

Molecular computing does not have an extensive background, as it is a fairly new research topic. Its origins, however, lie in the field of molecular electronics, where predominately chemists work to realize the behavior of wires, diodes, and transistors using only molecules. As such, this section provides a brief discussion on the progress of molecular electronics as it relates to molecular computing.

Beginning with the work of Arieh Aviram and Mark Ratner, molecular electronics has predominately been the result of chemists who have attempted to make molecules behave as electronic devices, such as diodes and transistors. From molecular wires to molecular logic gates, over the past thirty years, there has been significant progress in this field.

Although the prospect of molecular computing is bright, a few challenges must be overcome before it can be implemented successfully. First, the ability to fabricate many molecular logic gates in an orderly manner without the need to build molecule by molecule must be achieved. Second, these molecular logic gates must be able to interface with current computers in the macroscopic world. MEMS mechanical computing devices may very well be the answer to both of these issues. In fact, MEMS mechanical computing devices could be used to solve similar problems with biological computing systems as well.

2.4 Biological Computing

Current research in biological computing is focused on data storage or memory. Scientists are attempting to use deoxyribonucleic acid (DNA) for data memory that will last much longer than current memory devices [21]. Encoded messages can be produced using DNA strands which are implanted into host microorganisms. The two primary hosts are Escherichai coli (E. coli) and Deinococcus radiodurans (Deinorcoccus), since these microorganisms receive the embedded DNA quickly, can withstand harsh environments, and have high life expectancy.

The DNA encoding takes advantage of the four basic DNA building blocks, known as deoxyribonucleosides [21]. The four bases are Adenine (A), Cytosine (C), Guanine (G), and Thymine (T). Two complementary strands of DNA bond to one another to form a double-stranded DNA, where A bonds with T, and C bonds with G. Since there are only two base pairs, AT and CG, double-stranded DNA represents a digital sequence, which can be used to encode letters and symbols in much the same way that ASCII characters are encoded with 1's and 0's in computers.

Once the encoded DNA message is created and inserted into the host, it can be stored for a long period of time prior to extracting the information [21]. Current methods of message extraction are referred to as polymerase chain reaction, which requires that the DNA sequence be heated and cooled in cycles to amplify the DNA segment.

High volumes of data could potentially be stored using DNA, but many challenges exist. One challenge is the need to extract the information from the DNA quickly and efficiently. Just as with molecular computing, purely mechanical MEMS computing devices may serve as the necessary interface between the biological DNA memory system and the computer system in the macroscopic world.

Aside from the future prospects of biological computing, purely mechanical MEMS computing devices can also serve present needs, such as computing in harsh environments where electronic systems tend to fail. To understand this application, a brief discussion of harsh environments follows.

2.5 Harsh Environments

Harsh environments can be defined generally as conditions in which electronic circuitry are likely to behave in undesirable ways, resulting in poor system reliability. Two examples of harsh environments are extreme temperature and radiation.

2.5.1 High temperature effects on microelectronic devices. The temperature at which a microelectronic circuit operates greatly affects its behavior. Many of the physical properties of microelectronic devices change with temperature. Among these are bandgap energy, intrinsic carrier density, carrier concentrations, Fermi potential, and carrier mobilities [22]. Above 200 °C, present day Si devices tend to fail due to the change in electrical properties. Purely mechanical computing devices should be able to operate in much higher temperatures without failure.

Aside from operating reliably at high temperatures, there are many applications that require computers to operate in environments where radiation may present itself. The radiation environment and its effects on the electrical behavior of materials is discussed next.

2.5.2 Radiation effects on microelectronic devices. Radiation interacts with materials in three different ways. These are Rutherford scattering, photon interactions, and nuclear interactions [23]. Rutherford scattering, which is also referred to as Coulomb scattering, occurs between charged particles, and results in an energy loss due to ionization or photon emission [23]. Three different photon interactions are possible. First, low-energy photons can be absorbed by an atom, with the additional energy resulting in the ejection of an electron by the atom (photoelectric absorption) [23]. Second, photons can be scattered due to Coulomb scattering, with energy being transferred to a recoil electron [23]. Third, pair production can occur when a photon is absorbed by an atom, and an electron-hole pair is created [23]. Nuclear interactions are dominated by the elastic scattering of neutrons [23].

These radiation interactions result in two radiation effects that change the electrical behavior of semiconductor devices [24]. These are displacement, which refers to the dislocation of an atom within a crystal lattice, and ionization, which refers to the removal of an electron from an atom to form a free electron and an ionized atom [24]. The displacement and ionization effects cause three types of radiation damage to semiconductor devices, namely, displacement damage effects, transient damage effects, and surface damage effects [24]. Displacement damage effects are due to displacement, whereas the other two forms of radiation damage are caused by ionization [24].

Displacement damage effects are the changes in the electrical properties of a material that result from dislocations and physical imperfections to the crystal lattice structure [24]. Transient damage effects are the temporary changes in the electrical behavior of a device due to ionization of the bulk semiconductor material [24]. Surface damage effects are the more permanent changes in the electrical behavior of a device due to ionization near the surface [24].

Due to the presence of radiation in the space environment, the aerospace industry would benefit greatly from the development of computing devices that are reliable in the presence of radiation [25–27]. For electronic components in space, radiation can cause defects in the microelectronic material. These defects can cause "abnormal charge concentrations. Similar effects on semiconductor circuits range from a temporary change in logic state, because of the sudden local appearance of charge, to permanent substrate atom and charge dislocations that produce altered current-voltage characteristics and possible device failure" [25]. The process of hardening microelectronic components from radiation is an expensive process [26], which could be avoided by replacing the microelectronics with mechanical computing devices.

For applications where devices rarely operate in radiation environments, but may be required to on occasion, an alternative to completely replacing the current microelectronics would be to use the MEMS mechanical computing devices as back-up memory. That way, when the radiation environment is encountered, the mechanical computing device can store the current state of the microelectronic computer and this state can be restored to the computer once the radiation is no longer present.

With an understanding of the types of applications and environments in which MEMS mechanical computing devices are needed, the next logical step is to discuss MEMS, beginning with what they are and how they are fabricated.

2.6 Microelectromechanical Systems (MEMS)

Microelectromechanical Systems (MEMS) are small (on the order of 10-100 microns) systems that integrate mechanical components, sensors, actuators, and microelectronics on the same chip. Although not all of these components are present in every MEMS device, the possibility of having mechanical components that are controlled by microelectronic circuitry, all of which are fabricated using a similar process has lead to the growth in MEMS technology. The field of MEMS has its roots in microelectronics, and takes advantage of the processing that has been developed for microelectronic circuitry, but also branches out to a multitude of other disciplines, such as mechanical engineering, materials science, and chemistry. In reality, MEMS does not have a set definition. In essence, MEMS are small machines, or micromachines. The next section will discuss how they are made.

2.7 Microelectromechanical Systems (MEMS) Fabrication Processes

The fabrication of MEMS is typically performed in one of three ways. These are surface micromachining, bulk micromachining, and micromolding. Presented in this order, each of these processes is described in sufficient detail so that a reader unfamiliar with MEMS might understand the basics of MEMS fabrication.

2.7.1 Surface micromachining. Surface micromachining is a process by which a combination of sacrificial and structural layers are grown or deposited on a substrate and patterned, so that when the sacrificial layers are removed, the patterned structural layers are released, resulting in movable mechanical parts. The fabrication process of surface micromachined MEMS is similar to that of microelectronic circuit fabrication, in that thin films are deposited, patterned, and etched. Although Section 2.8 focuses on the surface micromachining fabrication process used in the design of the MEMS structures for this thesis, a detailed analysis of a generic surface micromachining process is described here, with Figure 2.9 used to help explain the process. The surface micromachining process begins with a substrate material

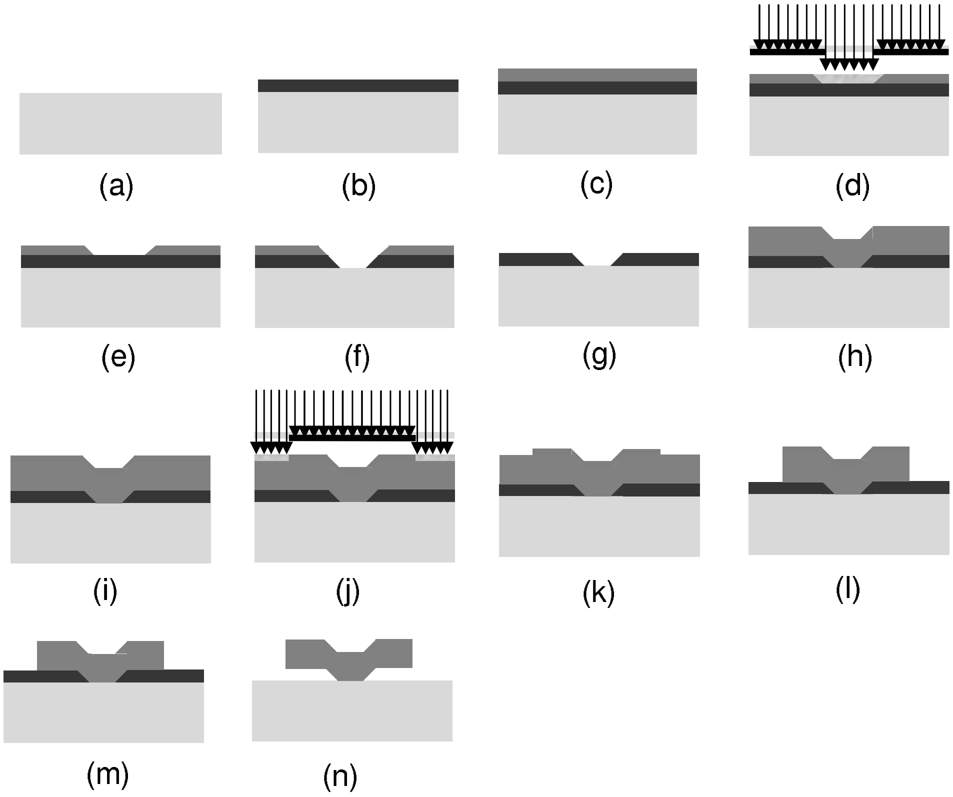


Figure 2.9 Surface Micromachining process. (a) Begin with a substrate. (b) Deposit a sacrificial layer. (c) Deposit a layer of photoresist (positive resist is shown). (d) Expose resist through a mask. (e) Remove exposed resist. (f) Etch the uncovered sacrificial layer. (g)Strip remaining resist. (h) Deposit a structural layer(conformal deposition is shown). (i) Deposit resist. (j) Expose resist through a mask. (k) Remove exposed resist. (l) Etch the uncovered structural layer. (m) Strip remaining resist. (n) Release structural layer by selectively etching sacrificial layer.

(Figure 2.9(a)). Next, a sacrificial layer is deposited (Figure 2.9(b)), followed by a layer of photoresist (Figure 2.9(c)). The photoresist is patterned with a masking step, and exposed by photolithography (Figure 2.9(d)). The exposed areas of the photoresist are now chemically altered, allowing them to be removed (assuming positive resist is used) with a chemical stripper (Figure 2.9(e)). If negative resist is used, the unexposed areas are removed by the stripper, and the exposed areas would not be affected. For this example, I assume positive resist is used. With the exposed resist removed, the sacrificial layer immediately below the removed resist is uncovered, and can be selectively etched (Figure 2.9(f)). The entire layer of photoresist can now be removed, completely uncovering the patterned sacrificial layer (Figure 2.9(g)). A structural layer is then deposited, and assuming a conformal process, the structural layer will follow the topography of the sacrificial layer (Figure 2.9(h)). Using resist, a masking step, and photolithography, the structural layer is patterned in much the same way as the sacrificial layer (Figure 2.9(i-m)). This process continues until the desired number of sacrificial and structural layers have been deposited and etched. Some common additions to this process include depositing metal and insulating layers to the structure. Both of these are used in the MUMPs® process, which is described in Section 2.8.

2.7.2 Bulk micromachining. While surface micromachining is an additive process, bulk micromachining is a subtractive process. In bulk micromachining, rather than adding pairs of sacrificial and structural layers and patterning them to form mechanical parts, structures are formed by etching into the substrate material. Using potassium hydroxide (KOH) as an anisotropic etchant for a silicon substrate material, certain three-dimensional structures are easily formed with bulk micromachining, "such as V-grooves, channels, pyramidal pits, membranes, vias, and nozzles" [28]. This facilitates the fabrication of many MEMS devices, such as accelerometers, pressure sensors, and motion sensors. Figure 2.10 shows a five-axis motion sensor that contains structures fabricated with bulk micromachining [29]. Figure 2.10(a) is a scanning electron micrograph (SEM) of the beam side of the structure, showing the beams and sensing plates that were formed by bulk micromachining of the first Si wafer. Figure 2.10(b) is an SEM of the mass side of the structure, showing the seismic mass that was bulk micromachined from a second

Si wafer after being fusion bonded to the first Si wafer, forming a Si-Si structure. Figure 2.10(c) is an optical image of the motion sensor mounted on a metal base. Figure 2.11 details the entire fabrication process of the motion sensor [29].

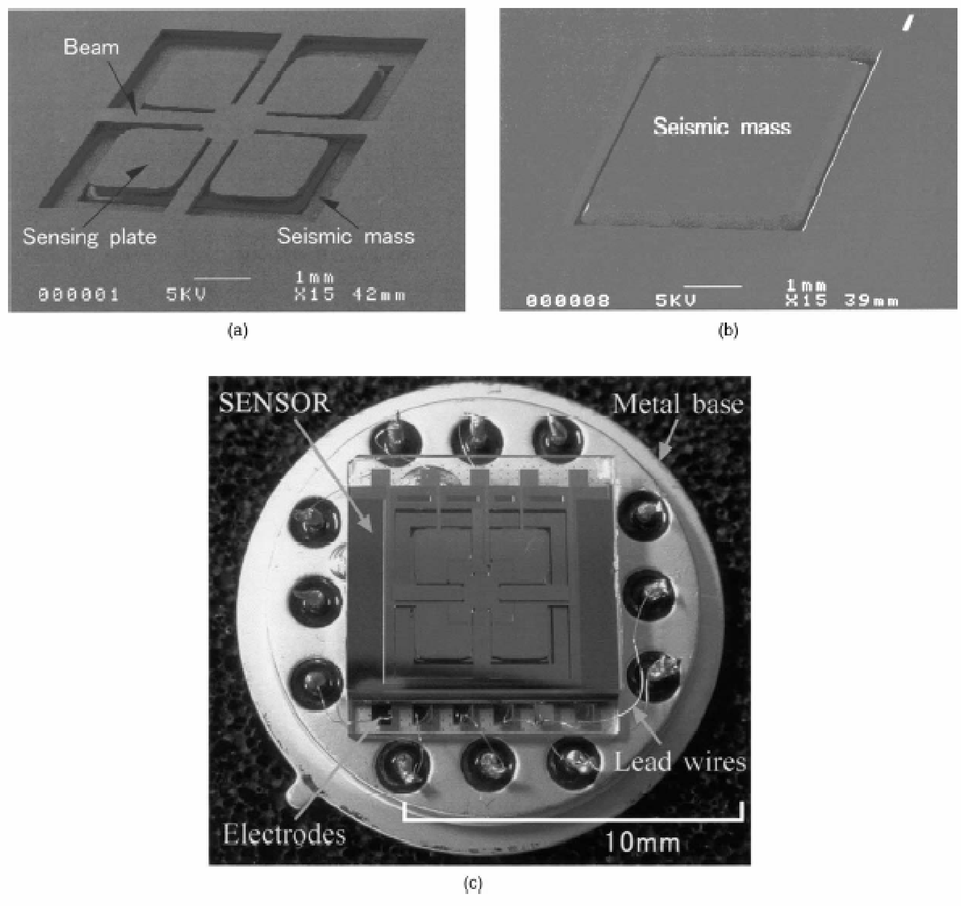


Figure 2.10 Five-axis motion sensor. (a) SEM of the beam side of the Si-Si structure. (b) SEM of the mass side of the Si-Si structure. (c) Sensor mounted on a metal base. [29]

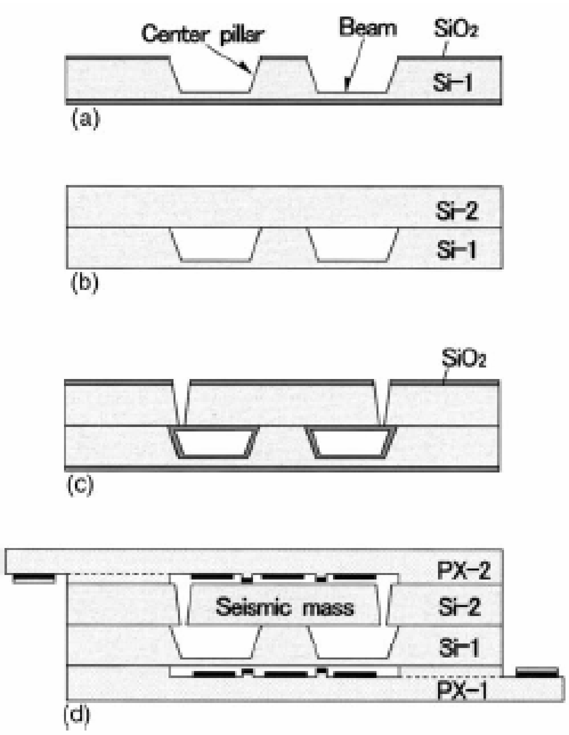


Figure 2.11 Fabrication process of five-axis motion sensor. (a) Bulk micromachine the beam structures. (b) Fusion bond a second silicon wafer. (c) Bulk micromachine the seismic mass structure. (d) Anodically bond two glass wafers to both sides of the structure. [29]

2.7.3 Micromolding. Micromolding is used for the formation of very precise structures that can be repeated with a high level of accuracy. The most common micromolding process is LIGA, which is a "German acronym for X-ray lithography (X-ray lithographie), electrodeposition (galvanoformung), and molding (abformtechnik)" [30]. In the LIGA process, a thick three-dimensional structure of X-ray resist is formed with a masking step involving X-ray exposure and developing [30]. This resist structure serves as a mold that is filled by metal electrodeposition [30]. The resist is then removed, leaving the metal structure, which can serve as either a reusable mold for plastic MEMS structures or as a final product [30]. Assuming the metal structure is to be used as a mold, plastic is injected into the metal mold to replicate the lost resist structure [30]. This new plastic mold can be reused to reproduce the metal structure that produced it [30]. The combination of this infinite molding loop with the highly precise reproduction of MEMS structures is the benefit of the micromolding fabrication process. Figure 2.12 shows a scanning electron microscope (SEM) image of a micromotor that was fabricated using the LIGA process.

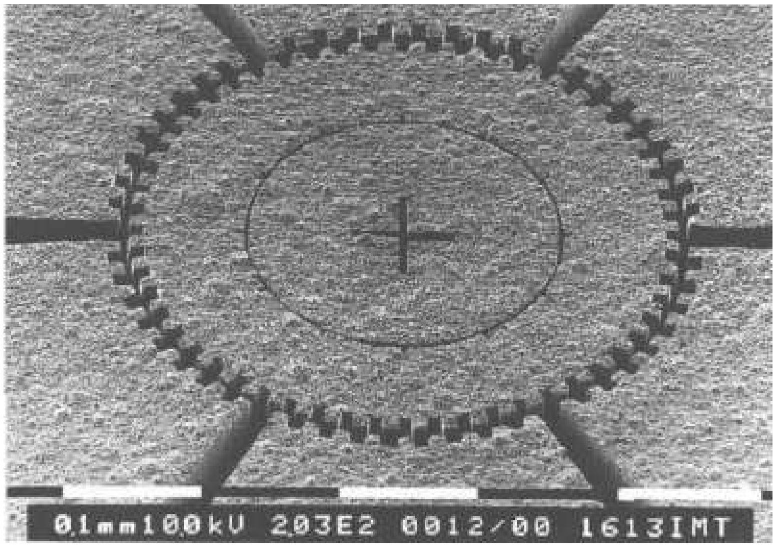


Figure 2.12 Electrostatic motor fabricated by the LIGA process. [31]

Due to the need for a fabrication process that caters to the needs of inexpensive prototyping, the Cronos PolyMUMPs foundry process was chosen to fabricate devices designed for this research. As such, Section 2.8 will give a detailed explanation of the PolyMUMPs processing steps. All processing steps are taken from the MUMPs® Design Handbook [12].

2.8 PolyMUMPs foundry processing steps

PolyMUMPs is a three layer polysilicon surface micromachining process that consists of two releasable structural polysilicon layers. Polysilicon serves as the structural layer and deposited phosphosilicate glass (PSG) serves as the sacrificial layer. An isolation layer of silicon nitride is deposited between the substrate and the first structural layer. The substrate is a 100 mm n-type (100) silicon wafer. Throughout this discussion, it may be beneficial to reference Figure 2.13, which details the layers and masking steps for the PolyMUMPs process. The silicon wafers are doped with phosphorous in a standard diffusion furnace using $POCl_3$ as the dopant source, which "helps to reduce or prevent charge feedthrough to the substrate from electro-

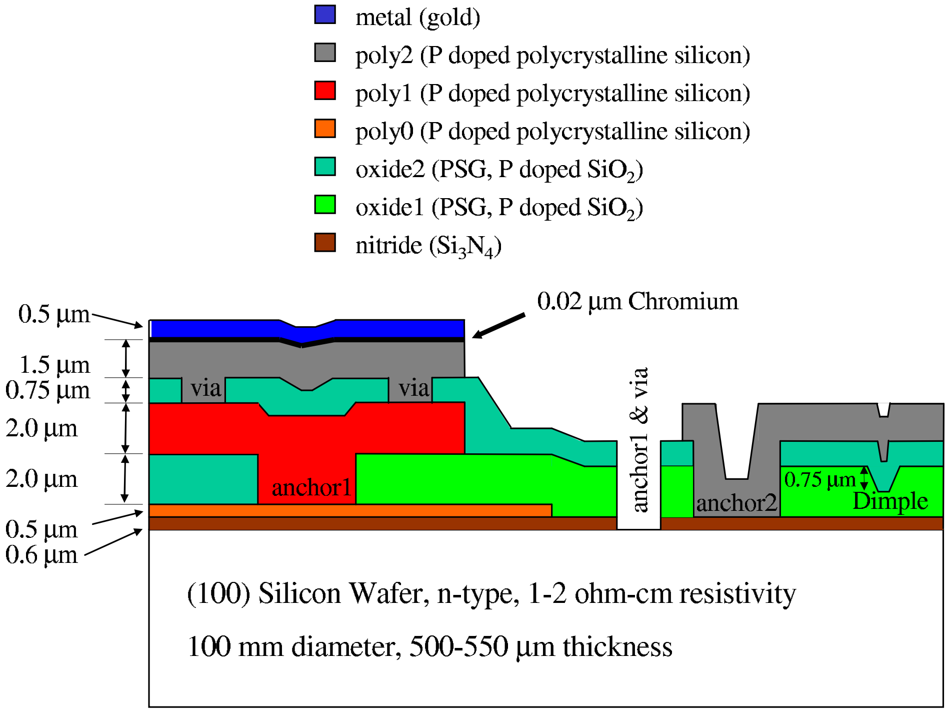

metal (gold)
poly2 (P doped polycrystalline silicon)
poly1 (P doped polycrystalline silicon)
poly0 (P doped polycrystalline silicon)
oxide2 (PSG, P doped SiO2)
oxide1 (PSG, P doped SiO2)
nitride (Si3N4)
0.5 µm
1.5 µm
0.75 µm
2.0 µm
2.0 µm
0.5 µm
0.6 µm
0.02 µm Chromium
via
via
anchor1
anchor1 & via
anchor2
0.75 µm
Dimple
(100) Silicon Wafer, n-type, 1-2 ohm-cm resistivity
100 mm diameter, 500-550 µm thickness

Figure 2.13 Layers and masking steps for the PolyMUMPs process.

static devices on the surface" [12]. A 600 nm insulating layer of silicon nitride is then deposited by low pressure chemical vapor deposition (LPCVD), followed by the deposition (LPCVD) of the first polysilicon structural layer (poly0).

The 500 nm thick poly0 layer is then photolithographically patterned and etched in a reactive ion etch (RIE) system. The first sacrificial layer, a 2 μm thick layer of PSG (oxide1) is then deposited by LPCVD and annealed at 1050 °C for 1 hour in argon [12]. Oxide1 is then patterned with a dimple mask, and 750 nm dimples are etched into the PSG by RIE. Oxide 1 is then patterned with the next mask, anchor1, and etched by RIE. This enables the first releasable layer, poly1, to be anchored to either the poly0 or nitride layer, depending on which layer is exposed.

A 2 μm thick layer of polysilicon (poly1) is then deposited by LPCVD, followed by a 200 nm thick layer of PSG. The wafer is then annealed at 1050 °C for 1 hour, causing the poly1 layer to be n-doped with the phosphorous that diffuses into it from both the thin layer of PSG above and the much thicker layer of PSG (oxide1) below. The polysilicon layer (poly1) and the thin layer of PSG are next patterned with the poly1 mask and etched to form the poly1 structural layer. After etching the poly1 structure, the photoresist that was used in the masking step is removed, followed by the removal of the PSG hard mask by RIE.

A 750 nm thick layer of PSG (oxide2) is deposited by LPCVD and annealed, as before. This second oxide layer is patterned using two different masks. The first mask step (via) is used to etch a via between the poly1 and poly2 structural layers, resulting in a mechanical and electrical connection. The second mask step (anchor2) is used to completely remove the oxide1 and oxide2 layers, allowing the poly2 layer to be anchored to the poly0 or nitride layer, just as anchor1 allowed for the anchoring of the poly1 layer. The etching for both mask steps is performed by RIE.

The second releasable structural layer, a 1.5 μm thick layer of polysilicon (poly2), is then deposited by LPCVD, followed by a 200 nm thick layer of PSG. The wafer is annealed at 1050 °C for one hour to dope the poly2 layer with phosphorous and harden the PSG layer which will be used as a hard mask. The poly2 layer is then patterned lithographically with the poly2 mask and etched by RIE, and the resist and PSG are removed.

The final layer that is deposited in the PolyMUMPs process is a 0.5 μm thick layer of gold, which is deposited and patterned using a metal lift-off technique.

With an understanding of the process used to fabricate the MEMS devices for this thesis, there is only one more item left to cover prior to unveiling the designs. Section 2.9 presents a detailed background of mechanical computing in MEMS, as this will allow the reader to gain an appreciation for the history behind this research.

2.9 Mechanical computing in Microelectromechanical Systems (MEMS)

Although they have served as inspiration for some of this research, the first MEMS mechanical computing devices were not purely mechanical. In reality, they were hybrid computing devices that incorporated mechanical movements with electrical inputs or outputs. A brief description of the different devices was given in the introduction. The following is a more detailed analysis of these MEMS mechanical computing devices.

Kruglick's MEMS relays were one of the first attempts at mechanical, rather than electrical switching [9]. The relays were designed in order to lessen the likelihood of device failure from that experienced by similar microelectronic (CMOS) transistor switches in high temperature or high radiation environments [9]. Kruglick's relays close a circuit by contacting a gold bar between two gold electrodes that are initially separated [10]. Figure 2.14 shows an SEM of the contact points for the microrelay. Switching for the relays is performed by applying power to arrays of electrothermal actuators to close the switch and removing power to open the switch [10]. Kruglick used his relays as a basic structure in the design of logic gates based on zero suppression (ZS) logic, but due to low yield, none of the devices were operational.

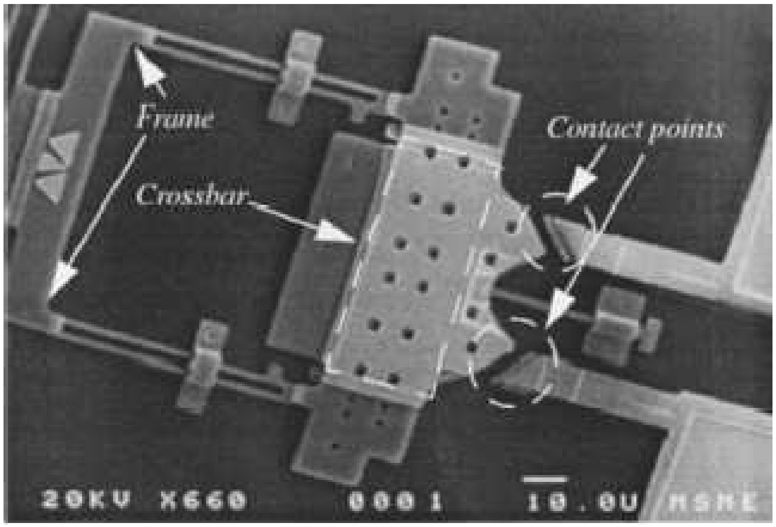


Figure 2.14 SEM of Kruglick's microrelay. [32]

Hirata's micromechanical switch was designed to be used as a logic element in multi-chip modules [7,8]. Figure 2.15 shows the basic switch design. One electrostatic mechanical switch was designed to perform the function of an AND gate, and another was designed for OR operation. Due to a difference in the geometry of the switches, the AND gate required both driving inputs to be on for the switch to latch. However, for the OR gate, only one of the driving inputs needed to be on to latch the switch [7,8]. Figure 2.16(a) shows an SEM of the switch used as an AND gate, with Figure 2.16(b) showing its switching behavior. The physical difference between the AND gate and the OR gate switches is that the AND gate beam is shorter than the OR gate beam, resulting in a higher elastic force. Consequently, the electrostatic force of one input is not high enough to close the AND switch, whereas the OR switch is closed with only one input. Figure 2.17 shows the switching behavior of the OR switch. Majority rule logic was also discussed as a possibility using Hirata's micromechanical switch by adjusting the drive electrode contact areas and beam lengths [7]. One of the main successes of Hirata's switch was that the voltage required for switching was reduced, without increasing the switch area [8].

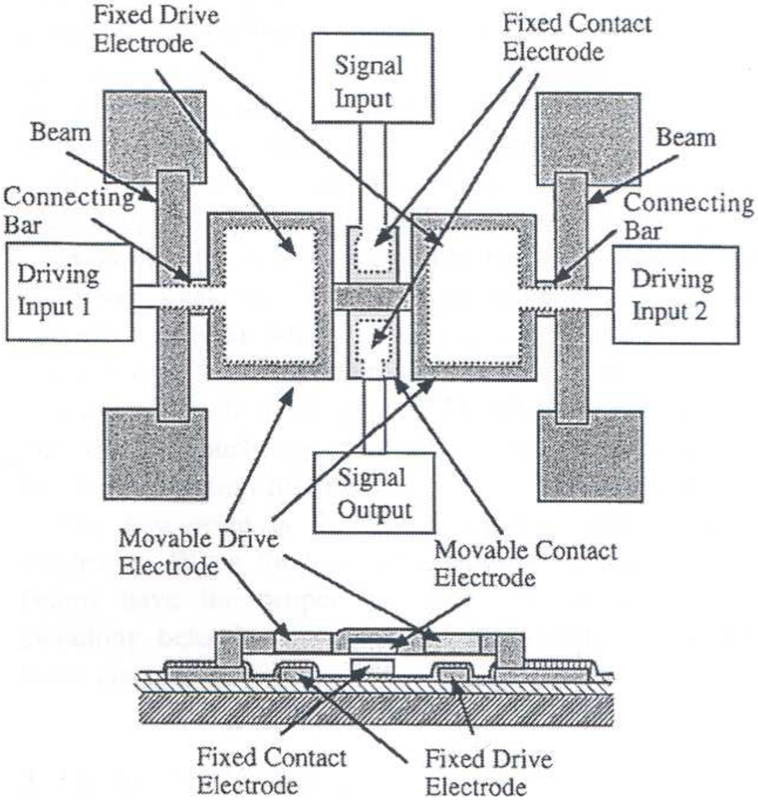


Figure 2.15 Schematic of Hirata switch design. [8]

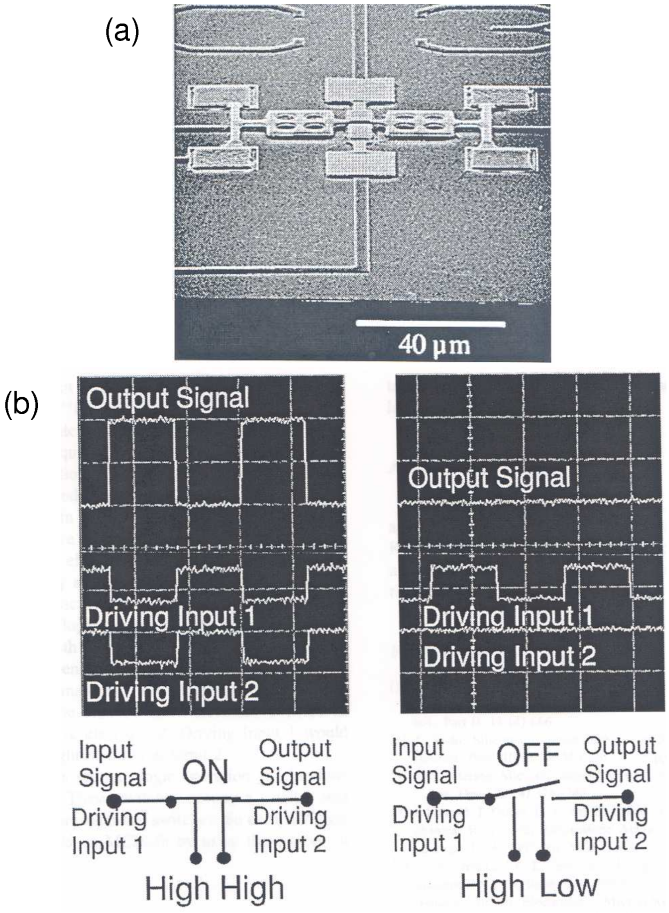


Figure 2.16 Hirata AND switch. (a) SEM of switch used for AND operation. (b) Switching behavior of AND switch. [8]

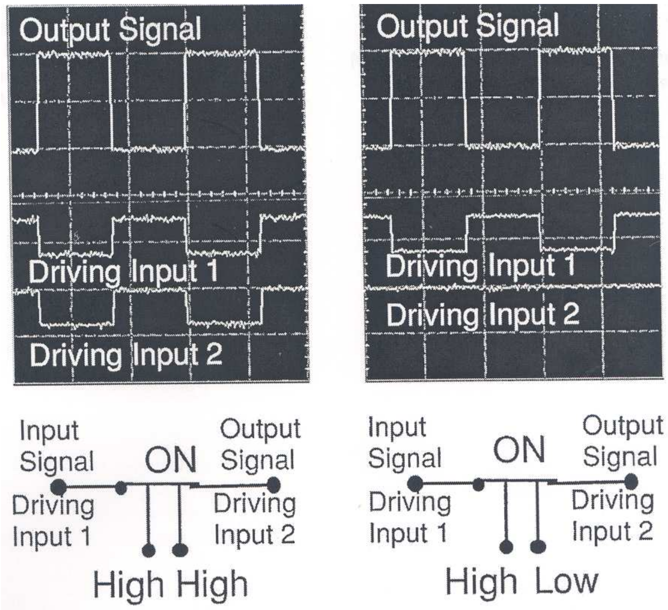


Figure 2.17 Switching behavior of Hirata OR switch. [8]

Another area of interest for mechanical computing involves mechanical digital-to-analog converters. Several articles have been published on this topic [3–6], which has resulted in two different designs.

Yeh's microelectromechanical digital-to-analog converter (MEMDAC) uses arrays of thermal actuators as its input, with a 5V source representing a binary 1, and no voltage representing a binary 0 [6]. With a voltage applied to a given bit, the corresponding lever arm is pulled into contact with the bumper, resulting in a movement in the output. The shift of the output due to a given bit being on is dependent upon the significance of the bit. The most significant bit (MSB) shifts the output half the distance of the lever arm movement. The next most significant bit shifts the output one-fourth the amount, and so forth. A four-bit MEMDAC is shown in Figure 2.18, but the design could possibly extend to an N-bit MEMDAC [6]. Some

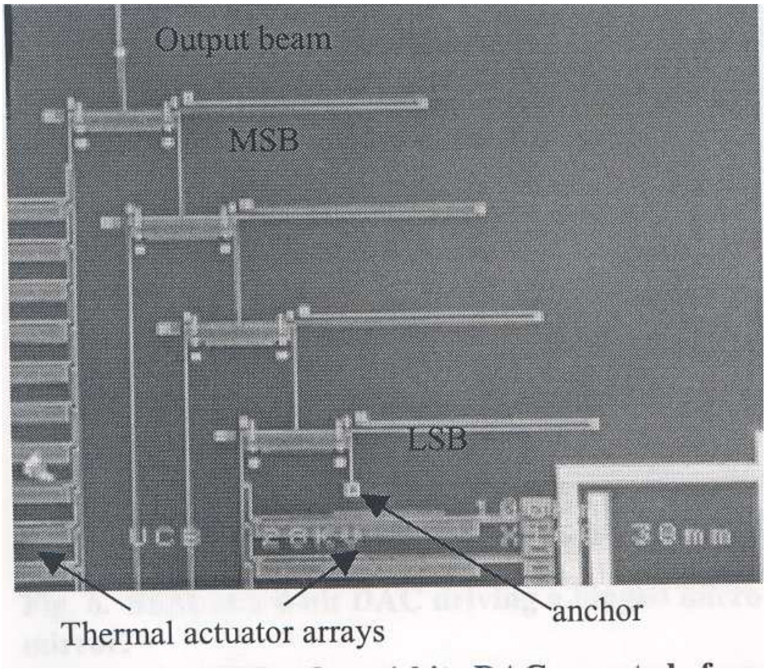


Figure 2.18 Yeh's Mechanical Digital-to-Analog Converter. [6]

areas in which this device could be used are “to drive mechanisms in micro-optics, mechanical computing, and micro-robotics” [6].

Toshiyoshi’s MEMDAC is based on the R-2R resistor network that is used for electrical digital-to-analog converters [3–5]. The mechanical equivalent is a C-2C compliance network, as shown in Figure 2.19(a). The MEMDAC was fabricated with “a set of cantilever suspensions and electrostatically-actuated shuttles”, as shown in Figure 2.19(b) [5]. Bit assignments are made by applying a voltage to a shuttle for a binary 1 and no voltage for a binary 0. As was the case with Yeh’s MEMDAC, the output movement that results from a given bit being turned on depends on the significance of the bit. The MSB causes the output to shift one-half the shuttle movement, the next most significant bit results in a shift of one-fourth, and so on. Toshiyoshi reported a 4-bit MEMDAC [3–5] which could be used in applications that require precise positioning, such as “the positioning mechanism on scanning probe microscopes” [5].

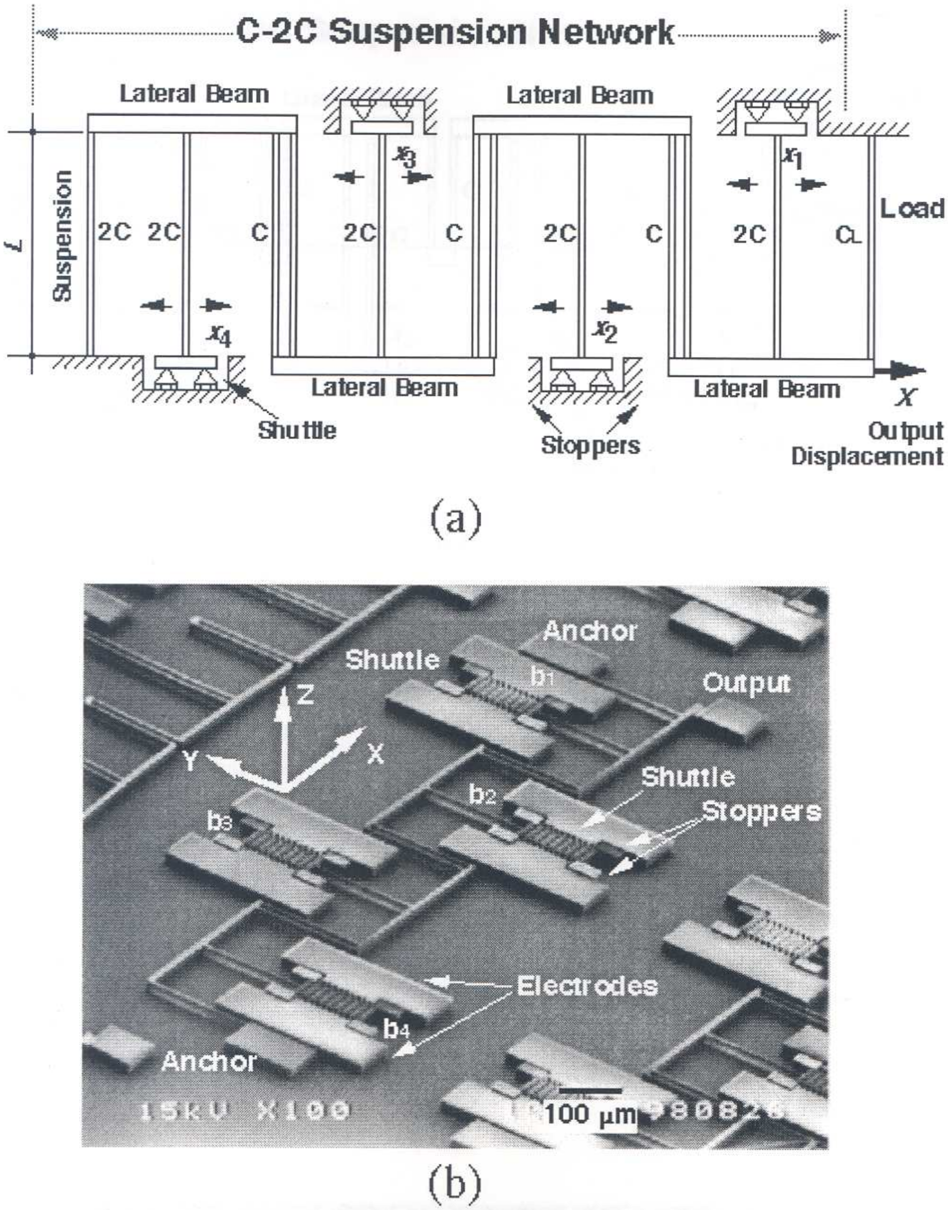
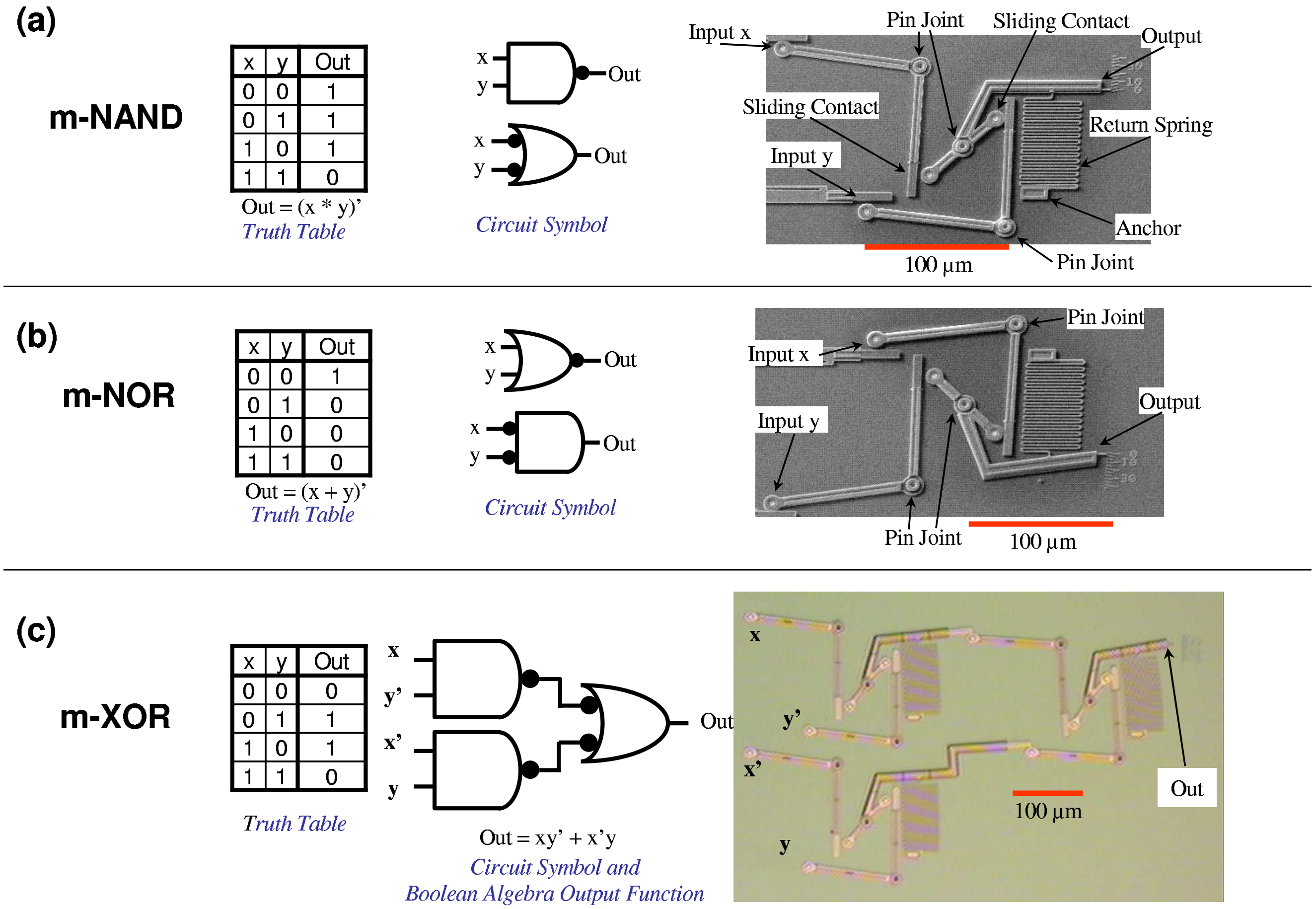


Figure 2.19 Toshiyoshi's Mechanical Digital-to-Analog Converter. (a) C-2C compliance network (b) SEM of MEMDAC [5]

The first purely mechanical computing components in the MEMS field were Kladitis' micromechanical logic gates [2]. Figure 2.20 shows Kladitis' NAND, NOR, and XOR logic gates. The purpose of his designs were three-fold. First, micromechanical computing devices have the potential for smaller size as compared to their microelectronic counterparts due to an inherent paradigm shift that occurs with the transition from microelectrical to micromechanical computing [2]. In the former, electrical input signals are processed by a microelectronic computer and electrical



x	y	Out
0	0	1
0	1	1
1	0	1
1	1	0

x	y	Out
0	0	1
0	1	0
1	0	0
1	1	0

x	y	Out
0	0	0
0	1	1
1	0	1
1	1	0

Figure 2.20 Kladitis' logic gates. (a) m-NAND gate. Circuit symbol, logic truth table, and SEM of the purely mechanical NAND gate (b) m-NOR gate. Circuit symbol, logic truth table, and SEM of the purely mechanical NOR gate. (c) m-XOR gate. Circuit symbol, logic truth table, and SEM of the purely mechanical XOR gate. Three m-NAND gates are used for the m-XOR gate. [2]

output signals are produced, with the microelectronic computer requiring a bulky power supply [2]. In the latter, mechanical input to mechanical output is achieved with a mechanical computer that does not require a power supply, resulting in a much smaller overall system [2]. Second, harsh environments, such as high temperatures and high levels of radiation have a damaging effect on microelectronic components, while mechanical devices would continue to operate reliably [2]. Third, micromechanical computing may be the next step towards quantum or molecular computing [2].

This research directly follows the work of Kladitis. Some designs are attempts at improving the devices found in his work. Others focus on novel analog computing devices. Chapter III presents these designs, along with their motivation and inspiration.

III. MEMS Device Design

This chapter discusses the designs for both analog and digital computing devices. Inputs and outputs are purely mechanical and all have been designed for the Cronos PolyMUMPs fabrication process [12]. Results are presented in Chapter VI for the devices that have been fabricated and tested, including the trigonometric function devices, a differential (adder), a multiplier, an integrator, and a digital-to-analog (D-to-A) converter. The tested devices come from PolyMUMPs runs 52,53,and 54 as shown in Figures A.4, A.5, and A.6, respectively. The differential, logic gates, and the analog-to-digital (A-to-D) converter are in the process of being fabricated as part of PolyMUMPs run 55 (Figure A.7). The testing of these devices is listed under future work.

3.1 Analog Computing

Purely mechanical analog computing devices convert continuous mechanical input motion to continuous mechanical output motion. The devices that were designed for this research that fall under this category of analog computing are the trigonometric functions (sine, cosine, and tangent/cotangent), integrators, multipliers, and differentials. All of these devices were fabricated and tested, with the exception of the differential, which is in the process of being fabricated at this time. The results from testing of an inferior differential design is presented in Section 6.2. Testing of the improved differential is listed under future work. This section presents the analog computing designs, along with their inspiration, and a derivation of their operation. The first devices that will be discussed are the trigonometric functions.

3.1.1 Trigonometric functions. Three different trigonometric functions were designed for this thesis. These are the sine, cosine, and tangent/cotangent functions. The designs for all three functions were inspired by the mechanical trigonometric functions that are found in [33]. With reference to the mechanical sine/cosine

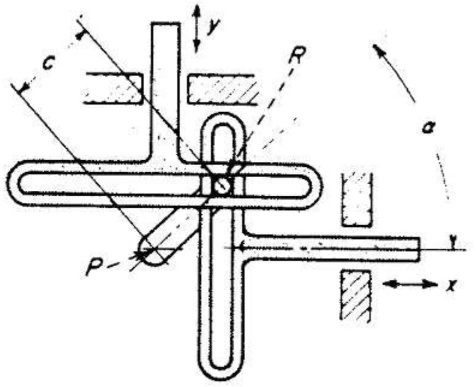


Figure 3.1 Diagram of the mechanical sine/cosine function [33].

function in Figure 3.1, the lever (of length c) rotates about the pivot, P. A pin, R, is connected to the lever and threaded through the T-shaped output sliders. With the lever rotated to a given angle, α, the slider outputs, x and y, have the following values:

$$x = c\cos(\alpha) \tag{3.1}$$

$$y = c\sin(\alpha) \tag{3.2}$$

As depicted in Figure 3.1, this combined function requires three movable, planarized layers. The PolyMUMPs process [12] only allows for two movable, conformal layers. Therefore, a new approach was undertaken to achieve these functions. First, the sine and cosine functions were implemented as separate devices. Second, in order to overcome the obstacle associated with conformal layers, the angle selector and output slider were fabricated separate from one another, as shown in Figures 3.2 and 3.3 for the sine and cosine functions, respectively. A simple assembly post-processing step is required after releasing the sine and cosine functions in order to position the devices for operation. This involves moving the output slider of each device so that the buckle encloses the knob, which is a poly1-poly2 stacked structure as shown in Figure 3.4. This is facilitated by the placement of a probe ring at the end of the output slider for positioning the probe tip while moving the slider. Also,

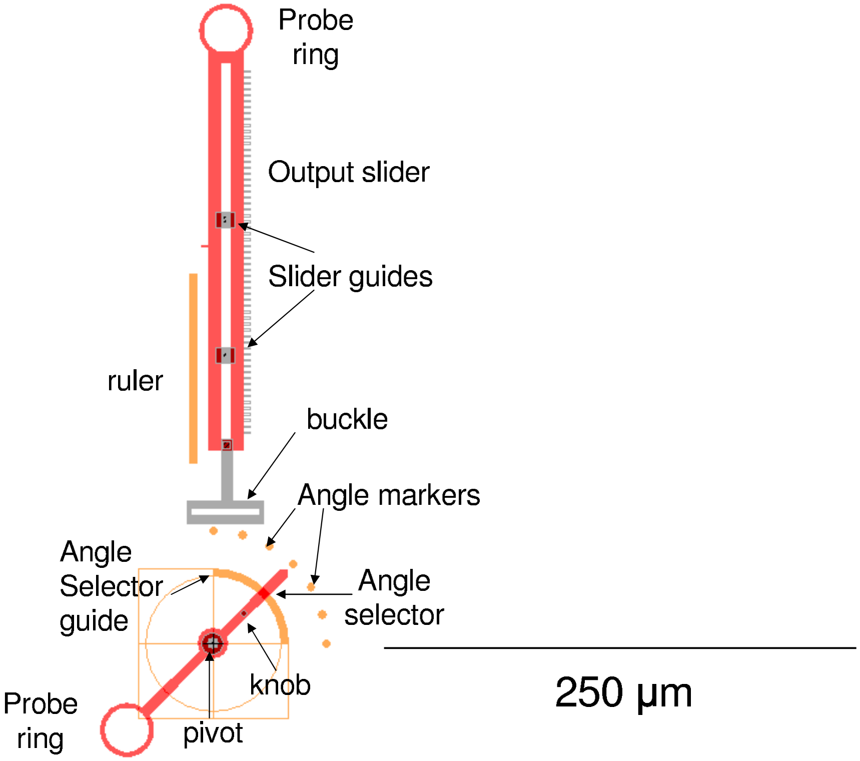


Figure 3.2 Computer aided drawing of the sine function in its as fabricated position with the output slider and angle selector separated.

poly2 guides have been positioned along the slider to confine the slider to purely bi-directional movement (neglecting the inherent backlash due to the limitations of the fabrication process). A probe ring has been placed at the end of the angle selector, as well. This is used to change the angle during device operation, but can also be used to ensure proper positioning of the angle selector prior to latching the buckle and the knob. Figures 3.5 and 3.6 depict the sine and cosine functions after device-specific assembly post-processing has been performed and they are ready for operation.

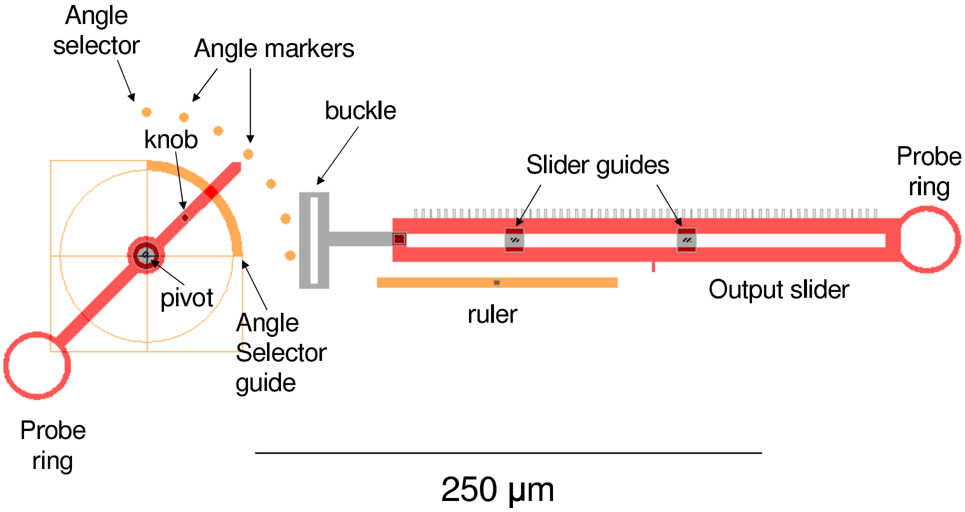


Figure 3.3 Computer aided drawing of the cosine function in its as fabricated position with the output slider and angle selector separated.

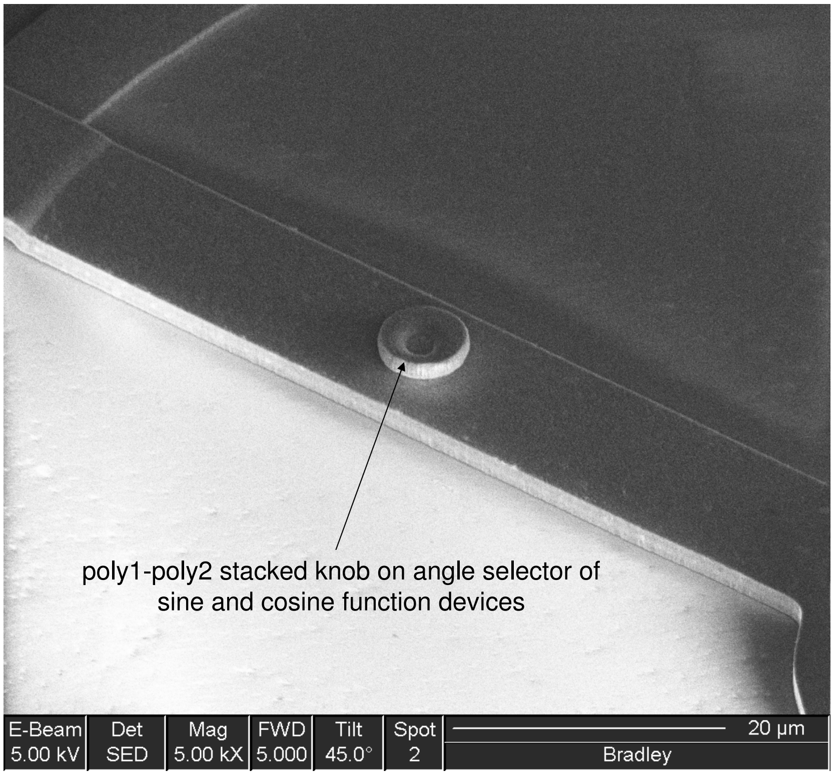

poly1-poly2 stacked knob on angle selector of sine and cosine function devices
E-Beam 5.00 kV
Det SED
Mag 5.00 kX
FWD 5.000
Tilt 45.0°
Spot 2
20 μm
Bradley

Figure 3.4 Close-up of poly1-poly2 knob on the angle selector for the sine and cosine function devices.

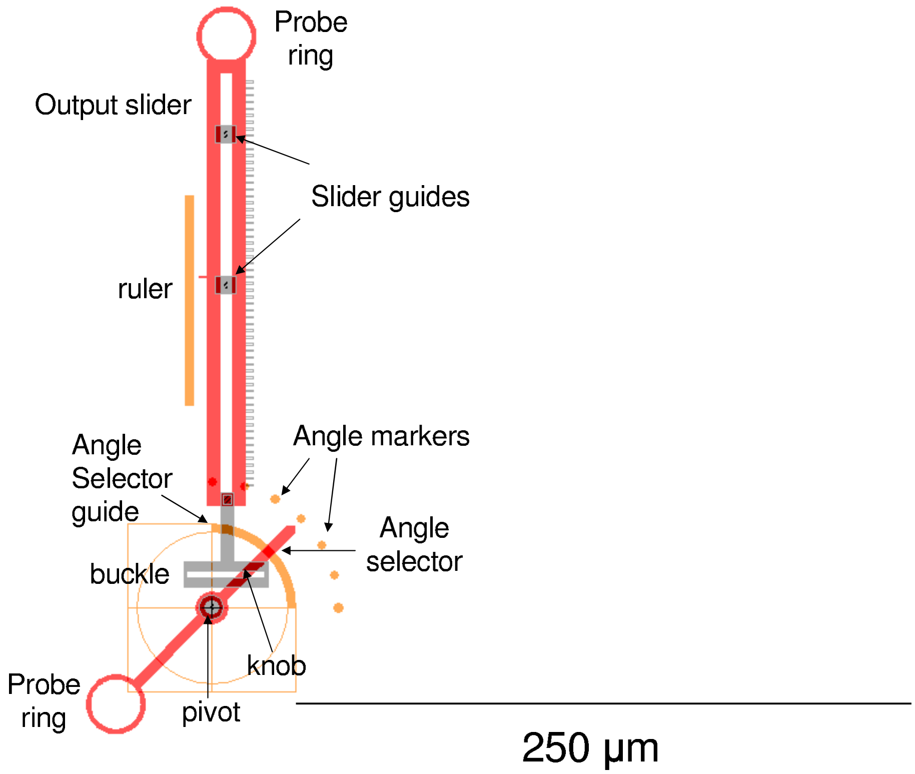


Figure 3.5 Computer aided drawing of the sine function after assembly post-processing is performed to position the buckle so that it encloses the knob.

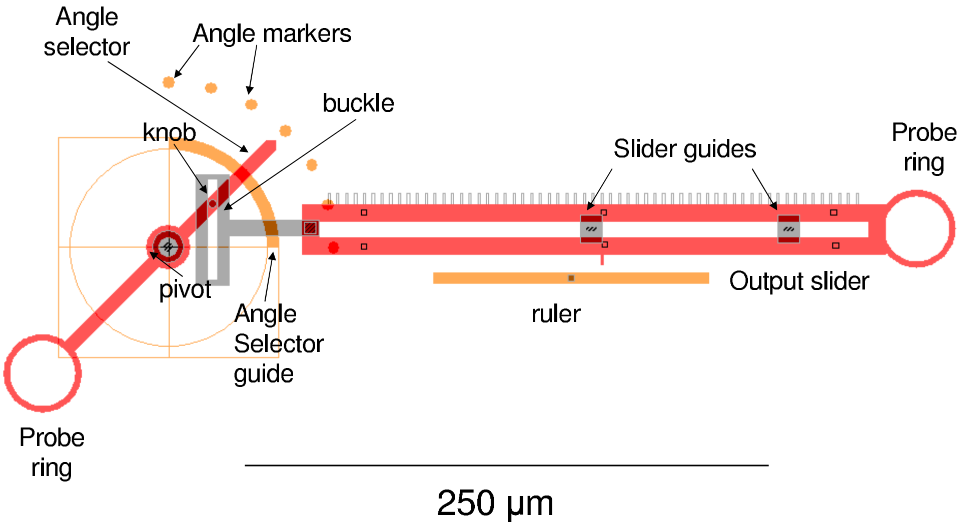


Figure 3.6 Computer aided drawing of the cosine function after assembly post-processing is performed to position the buckle so that it encloses the knob.

As fabricated, the angle selector for both functions is initially set to an angle of 45°, and the knob is positioned a distance of 56.5 μm from the pivot point. By sliding the output slider so that the buckle encloses the knob, the output value that is read from the ruler is 40, corresponding to the distance (in μm) from the position of the output marker when an input angle of 0° is selected to the marker position when an input angle of 45° is selected. This corresponds to the result of the true sine and cosine functions for these settings, which are 56.5 sin(45°) = 56.5 cos(45°) = 40.

There are two key drawbacks to these devices. First, the separation of the angle selector and output slider upon fabrication results in an excess chip area required to fabricate the smaller working device. Second, it may be difficult to perform the required assembly post-processing of sliding the poly2 buckle over the knob without inadvertently changing the position of the angle selector. Both of these drawbacks could be avoided by using a different fabrication process, such as Sandia's SUMMIT V process, which has four releasable layers, two of which are planarized [34]. This would allow for the fabrication of a combined sine/cosine function as depicted in Figure 3.1.

The other trigonometric function that was designed for this thesis is the tangent/cotangent function. The inspiration behind this design comes from the mechanical trigonometric function found in [33] and shown in Figure 3.7. With one output slider, the mechanical tangent/cotangent function performs both operations. This results from the fact that the tangent of an angle (α) is equal to the cotangent of its complement (β = 90 ° - α) for angles between 0° and 90°. As drawn in Figure 3.7, the output is expressed as

$$x = c\tan(\alpha) = c\cot(\beta) \tag{3.3}$$

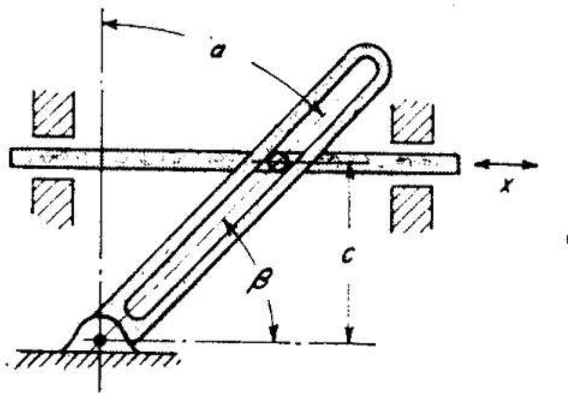


Figure 3.7 Diagram of the mechanical tangent/cotangent function [33].

where α and β are the complementary angles shown in the figure, and c is the vertical distance from the pivot point to the center of the slider. The following is a derivation of this function. Figure 3.8 shows a simplified, labelled drawing of the right triangle formed by the angle selector and the output slider on the top left side of the angle selector of Figure 3.7. With respect to this figure, the output, x, is

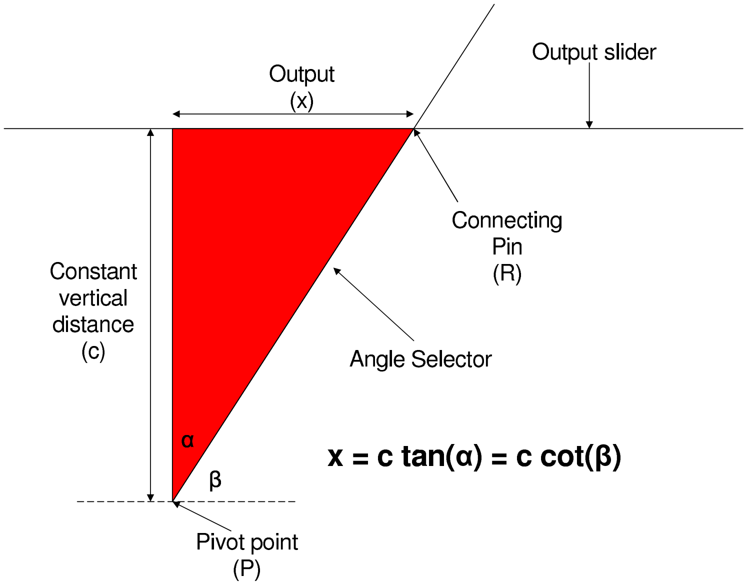


Figure 3.8 A simplified, labelled drawing of the right triangle formed by the angle selector and output slider of Figure 3.7.

equal to the horizontal distance between the pivot point and the pin that connects the output slider to the angle selector. The angles are labelled α and β, and the vertical distance between the pivot point and the output selector is the constant value, c. The resulting value of the output to this function corresponds to Equation 3.3. The second part of Equation 3.3 holds, since α and β are complementary first quadrant angles.

As with the sine and cosine functions, in order to fabricate the tangent/cotangent function using the PolyMUMPs process, the angle selector and output slider were fabricated separate from one another, as shown in Figure 3.9. A simple assembly

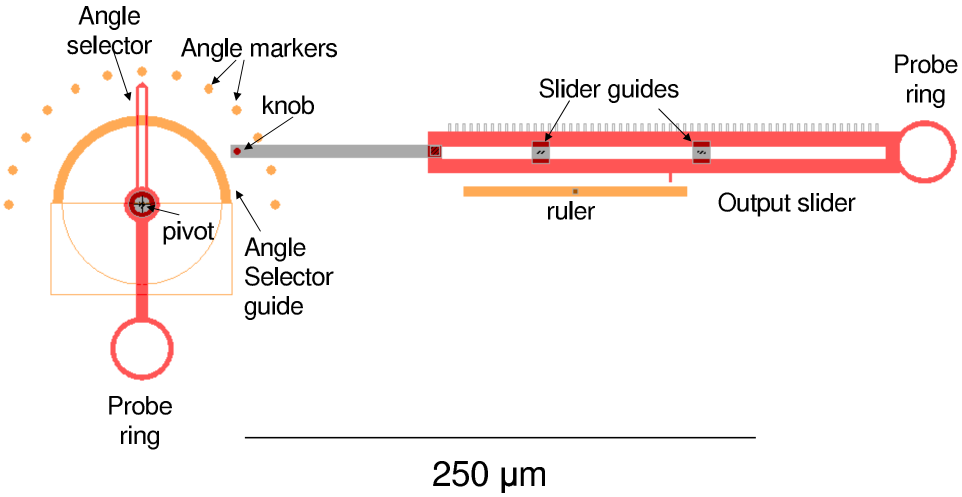


Figure 3.9 Computer aided drawing of the tangent/cotangent function.

post-processing step of sliding the output slider so that the knob is confined to the slot on the angle selector is required before the tangent/cotangent function can be operated. Figure 3.10 shows the result of this post-processing. The initial inputs for this function are $\alpha = 0°$ and $\beta = 90°$, with the constant value of c being 60 μm. The initial output value corresponding to the initial device settings is 0. As with the sine and cosine functions, the output to the tangent/cotangent function is read as the distance from the position on the poly0 ruler that is pointed to by the output marker for an input angle of $\alpha = 0\ °$ to the location pointed to by the output marker when the angle selector is set to the desired input angle.

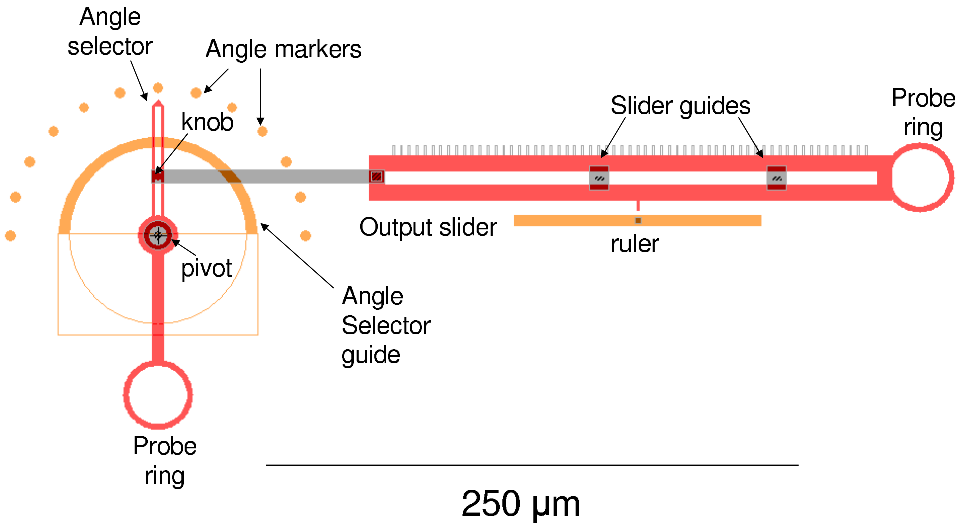


Figure 3.10 Computer aided drawing of the tangent/cotangent function after assembly post-processing has been performed so that the knob is enclosed by the slot on the angle selector.

3.1.2 Differential. A differential is used to add or subtract two values. The differential design for this thesis is inspired by the sliding-link differential shown in Figure 3.11 [33]. Two input sliders are used to represent the numbers to be added. Moving a slider up from its zero location represents a positive number. Moving a slider down represents a negative number. The output slider is linked to a crossbar that connects the two inputs to the output at pivot points, as shown in Figure 3.11 [33]. In general, the output to the sliding-link differential is

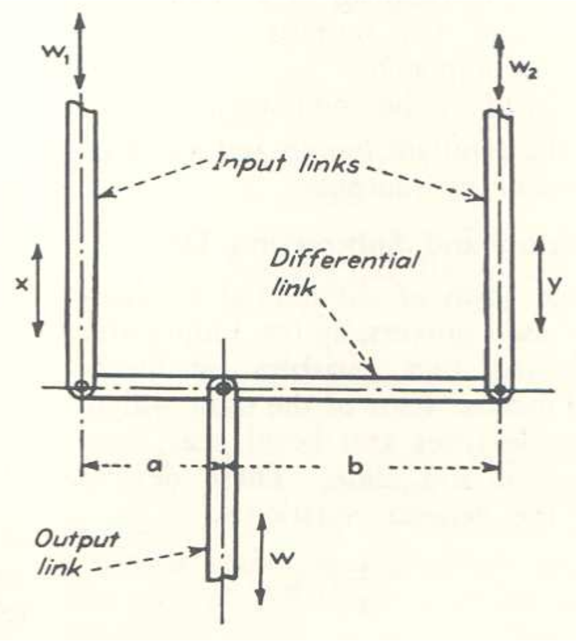


Figure 3.11 Diagram of a sliding-link differential [33].

$$W = \frac{bX + aY}{a + b} \tag{3.4}$$

where W is the output value, X and Y are the two inputs, and a and b are the crossbar parameters, as shown in Figure 3.12. The following is a derivation of this function. The influence of input X on the output, W, is

$$W_1 = \frac{bX}{a + b} \tag{3.5}$$

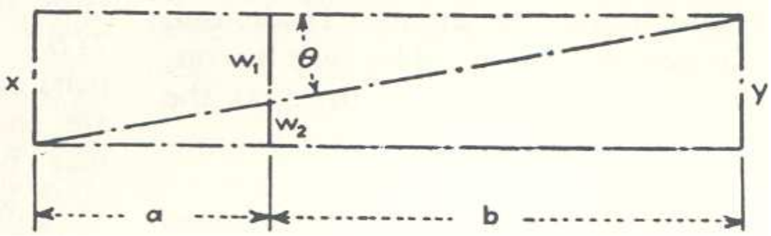


Figure 3.12 Sliding-link differential parameters [33].

Similarly, the influence of input Y on the output, W, is

$$W_2 = \frac{aY}{a+b} \tag{3.6}$$

Combining terms, the output is defined as the sum of W_1 and W_2, which gives:

$$W = W_1 + W_2 = \frac{bX + aY}{a+b} \tag{3.7}$$

Consequently, this is the same result given in Equation 3.4. With a careful choice of parameters, i.e. setting a and b equal, this equation reduces to:

$$W = \frac{X+Y}{2} \tag{3.8}$$

The output is equal to the average of the two input values. Multiplying this result by 2 gives the true output to an addition of the input terms. This is accomplished with a gear ratio of 2 at the output, as shown in Figure 3.13. The entire differential design is shown in Figure 3.14, with a close-up of the pivot connections shown in Figure 3.15.

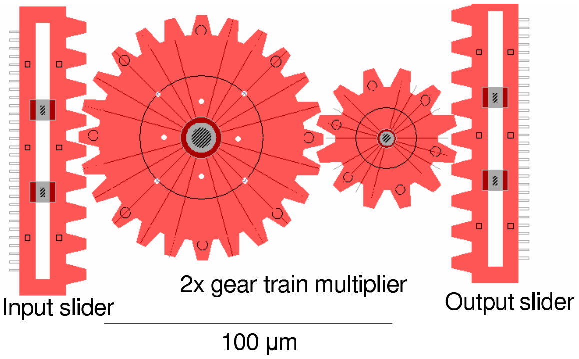


Figure 3.13 Computer aided drawing of the output gear train for the sliding-link differential. The gear train multiplies the output of the function by 2.

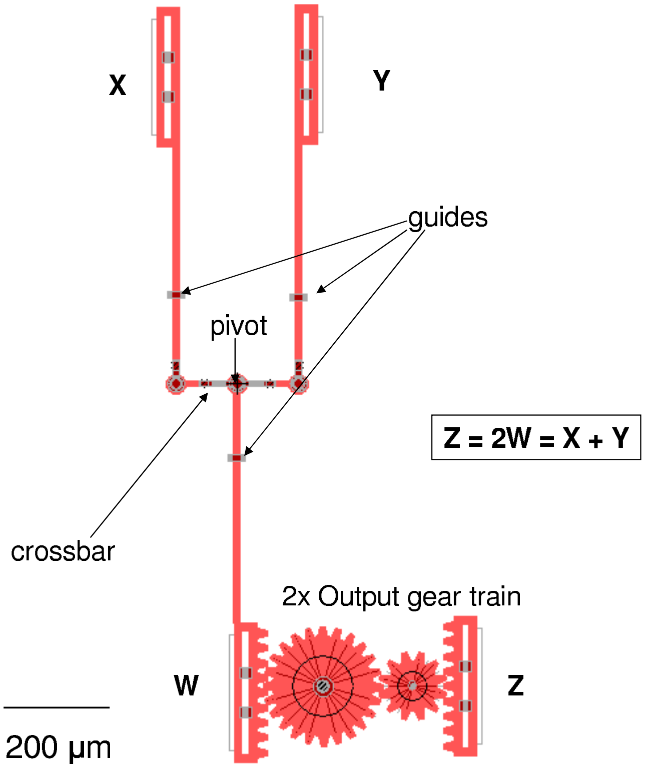


Figure 3.14 Computer aided drawing of the sliding-link differential.

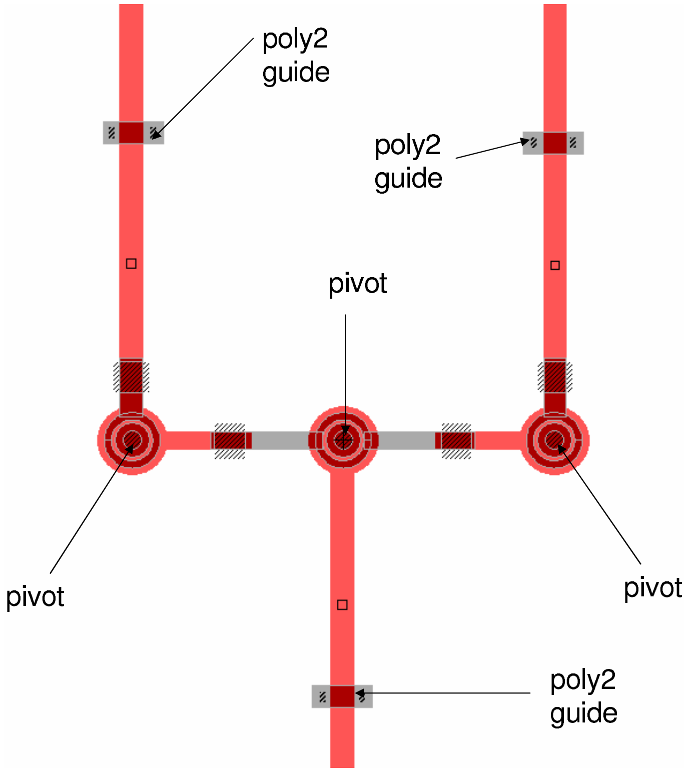


Figure 3.15 Computer aided drawing of the pivot connections for the sliding-link differential.

3.1.3 Multipliers. The multiplier is used to multiply two numbers. The multiplier design is based on the similar triangle multiplier found in [33] and shown in Figure 3.16. It takes advantage of the geometric properties of similar triangles in order to allow for multiplication. Figure 3.17 is used to help explain the operating principle behind the multiplier. From the geometric properties of similar triangles,

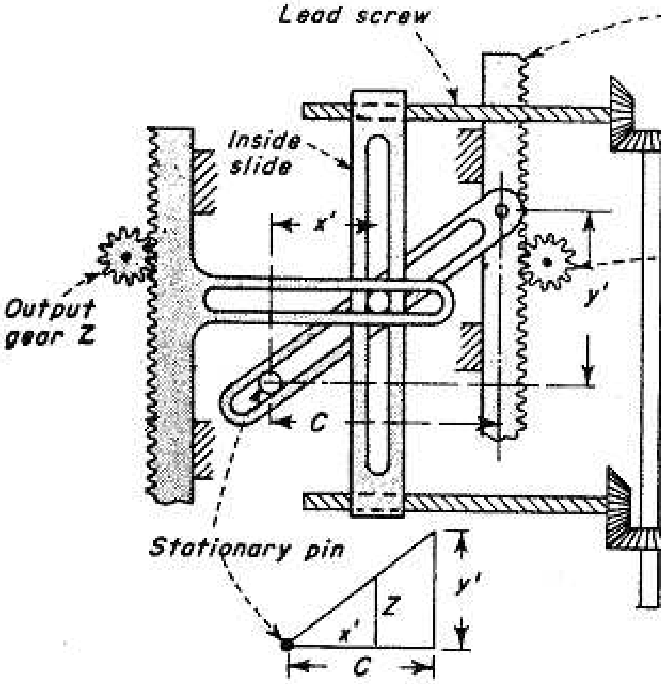


Figure 3.16 Diagram of a mechanical similar triangle multiplier [33].

$$\frac{Z}{Y} = \frac{X}{c} \tag{3.9}$$

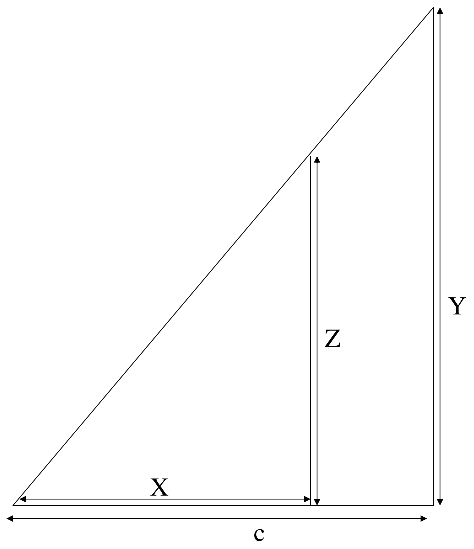


Figure 3.17 Similar triangle properties.

Rearranging terms gives

$$Z = \frac{XY}{c} \tag{3.10}$$

The output, Z, is equal to the product of X and Y divided by the constant scaling factor, c. Figure 3.18 shows the design for the multiplier used in this thesis, with Z, X, Y, and c as specified above. For this device, the conformal polysilicon layers of the PolyMUMPs process were used to connect the poly2 inputs and output to the poly1 diagonal bar which is free to pivot about the pivot point. The value of x is adjusted by sliding the horizontal slider, and the y value is adjusted by sliding the vertical slider. The poly2 guides on the input sliders restrict the x and y input sliders to purely horizontal and vertical movement, respectively. The x value corresponds to the horizontal distance from the pivot to the output bar, the y value corresponds to the vertical distance between the pivot and the point at which the y input slider crosses the diagonal bar, and the output is the distance between the bottom of the vertical poly0 output ruler and the end of the output bar. Moving the y input slider in the direction indicated by the arrow in Figure 3.19 causes the conformal poly2 layer of the slider at the poly2/poly1 conformal connection to move the poly1

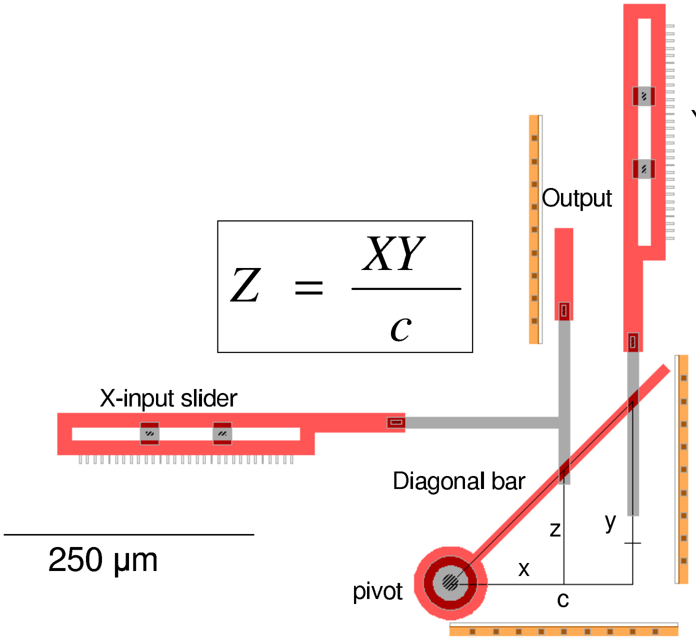


Figure 3.18 Computer aided drawing of the multiplier.

diagonal bar to a higher angle. This results in a vertical movement of the output slider, as indicated by the arrow next to the output slider. Moving the x input slider in the direction shown also causes the output slider to move up, but this is due to the fact that the output slider moves along the length of the diagonal bar.

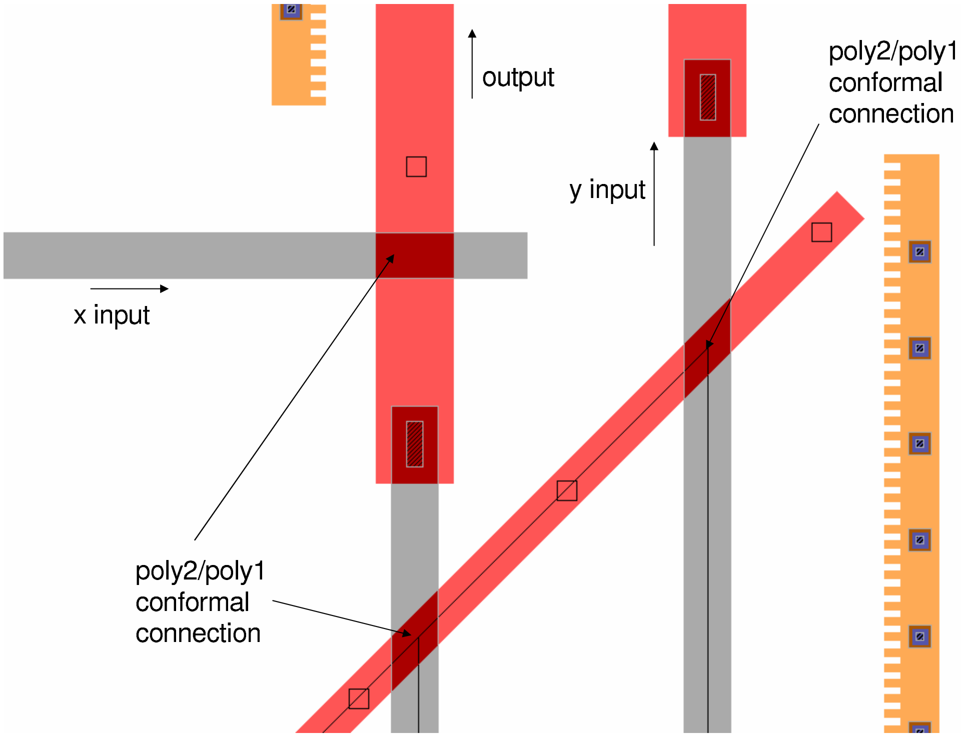


Figure 3.19 Computer aided drawing of the poly2/poly1 conformal connections on the multiplier.

3.1.4 Integrators. MEMS integrators have been used in such applications as safe and arming devices. The design of the mechanical integrator that is used in this thesis is based on the disk integrator from [33]. The disk integrator, as shown in Figure 3.20, performs the following calculation:

$$dZ = (1/c)YdX \tag{3.11}$$

where dX is an infinitesimal rotation of the disk, Y is the distance from the center of the disk to the friction wheel, c is the radius of the wheel, and dZ is the corresponding rotation of the output gear. Expressed in integral form, equation 3.11 becomes:

$$Z = (1/c)\int YdX \tag{3.12}$$

The derivation of this function is as follows: Input X rotates an infinitesimal angle dX, which causes a point on the disk a distance Y from the center to rotate a distance YdX. This, in turn, moves the friction wheel the same distance, causing an output rotation, dZ of $\frac{1}{c}Ydx$.

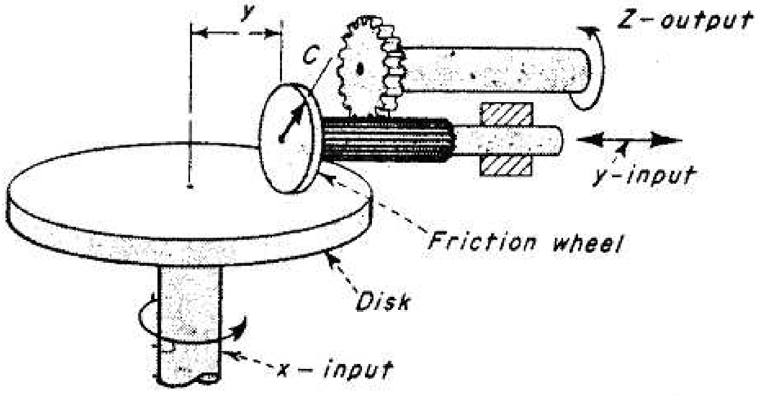


Figure 3.20 Diagram of a disk integrator [33].

Figure 3.21 shows the design of the integrator for this thesis. It operates based on the principles described previously with reference to the disk integrator. In order

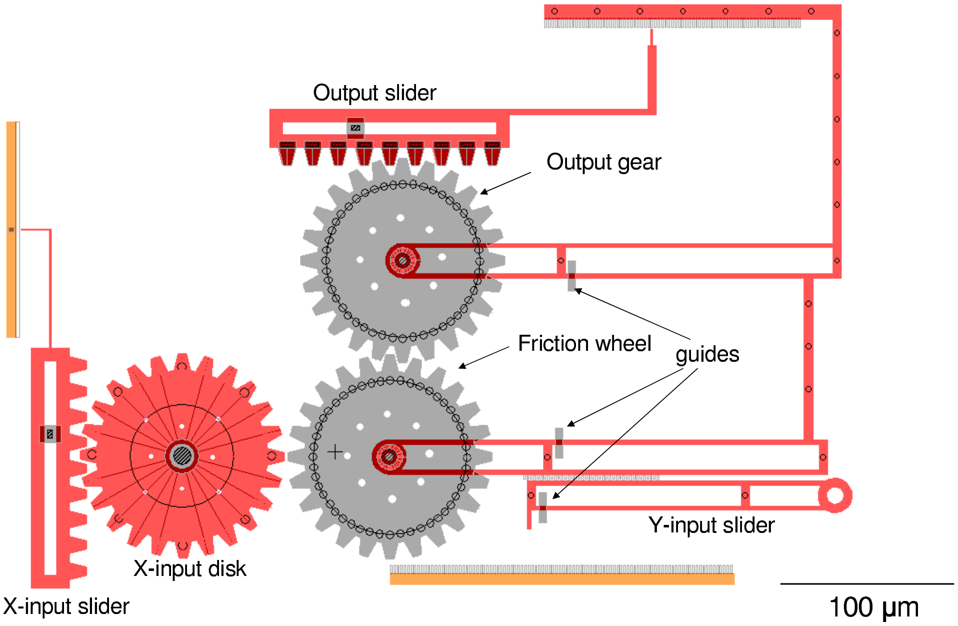


Figure 3.21 Computer aided drawing of the integrator in its as-fabricated position with the friction wheel and X-input gear separated from one another.

to be fabricated in the PolyMUMPs process, which only has two movable structural layers, dimples were used on the poly2 gear to represent the friction wheel, and the structure containing the friction wheel, output gear, output slider, and output ruler are fabricated separate from the x-input disk, as shown in Figure 3.21. Sliders are used to position the friction wheel (set Y) and rotate the x-input disk (set X). Positioning the friction wheel, as shown in Figure 3.22, is required before the device is operational. Poly2 guides are used to restrict the slider and structural movements. The output gear and friction wheel are fabricated as poly2 gears with poly1 hubs and small poly1-poly2 stacked pins. For this design, c is equal to the radius of the input disk. The output slider is moved by the rotation of the output gear.

This concludes the presentation of analog computing device designs. Results for operational testing of these devices is contained in Chapter VI. The designs for digital computing devices is discussed now.

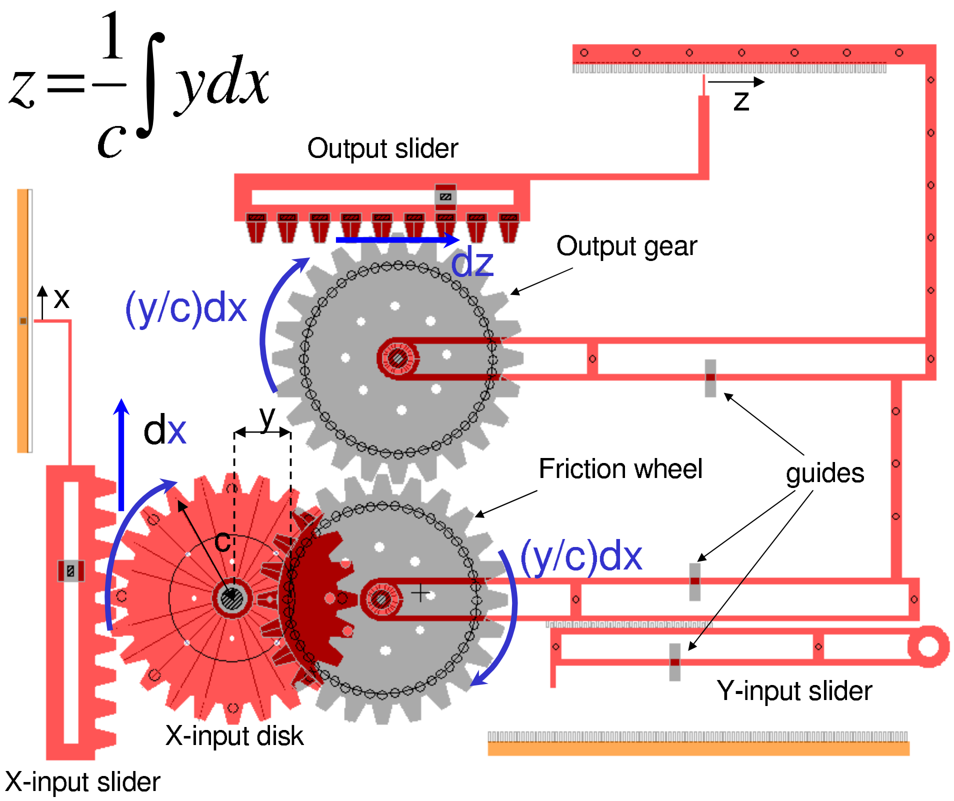


Figure 3.22 Computer aided drawing of the integrator after assembly post-processing is performed to place the dimples of the friction wheel on top of the X-input disk at a distance Y from the center of the disk.

3.2 Digital Computing

Designs of mechanical digital computing devices are presented in this section. As with the mechanical analog devices, this section discusses the inspiration behind the designs. The designs are presented, along with an accompanying explanation of their operation. Switching speed and force requirements are discussed for each of the digital computing devices in Chapter IV.

For purely mechanical digital computing, the initial state of the computing device is set when the device is fabricated. This differs from common microelectronic logic gates, whose initial states are not determined by their physical structure. Consequently, every mechanical digital computing device has N distinct physical structures, where N is the number of possible input combinations. For example, an inverter, which has two possible inputs, has two different structures. A logic gate, on the other hand, has four different structures, since each has two inputs and four different input combinations.

Only the inverter has been fabricated and tested, so the discussion of this design appears first. It is followed by a discussion of the design of NAND, NOR, and XOR logic gates. The testing of these devices is listed under future work in Section 7.5.

3.2.1 Inverters. One of the most basic digital computing devices is an inverter. There are several ways to implement this function, including levers and gears. The design chosen for this thesis uses a gear with an input and output slider. Figure 3.23 shows the design of the inverter with the input initially set to 0. Figure 3.24 shows the design of the inverter with the input initially set to 1. As its name suggests, an inverter sets the output opposite of the input. When the input slider of Figure 3.23 moves up, the output slider moves down. For mechanical computation up and down represent binary values 1 and 0, respectively. Table 3.1 defines the logic operation of an inverter.

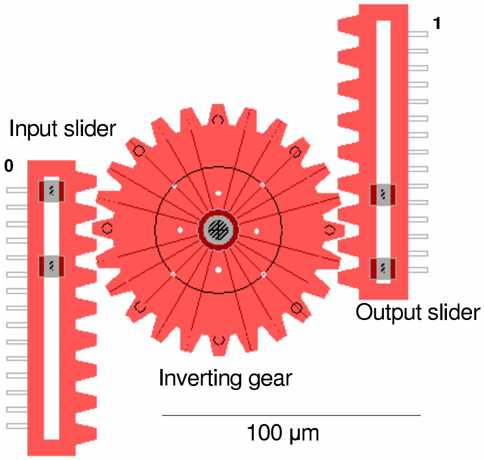


Figure 3.23 Computer aided drawing of the inverter with its input initially set to 0.

Input	Output
1	0
0	1

Table 3.1 Inverter truth table.

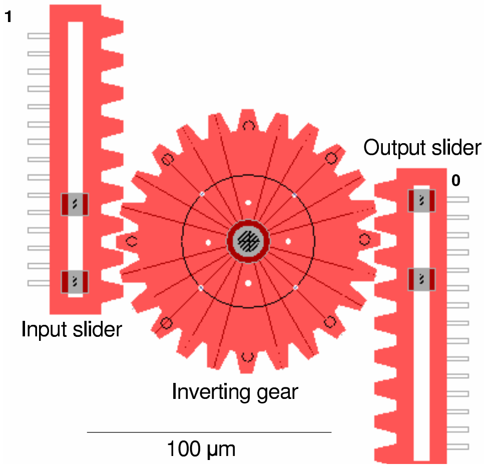


Figure 3.24 Computer aided drawing of the inverter with its input initially set to 1.

3.2.2 NAND gates. The NAND gate is a digital computing device that operates according to the truth table found in Table 3.2. This design was based on Kladitis' NAND gate from [2], and is an attempt to improve upon it. Sliders are used at the inputs and output, gears are used in place of levers, and all four physical gate configurations are presented.

Input 1	Input 2	Output
0	0	1
0	1	1
1	0	1
1	1	0

Table 3.2 NAND truth table.

Two distinct NAND gate structures are used, one for the output initially set to 1 (NAND-1) and the other for the output initially set to 0 (NAND-0). This is necessary, since restoring springs are used in connection with the output to the logic gates. Every logic gate structure has a relaxed state and a stretched state, corresponding to the state of the restoring spring. The device must be fabricated in the relaxed state. Since the inputs do not have restoring springs, they can be fabricated in either state. However, for the NAND gates to behave as depicted in Table 3.2, the NAND-1 structure is used for the first three input combinations. These are the NAND-00, NAND-01, and NAND-10 configurations, as shown in Figures 3.25, 3.26, and 3.27, respectively. The NAND-0 structure has only one configuration, which is referred to as the NAND-11 structure, as shown in Figure 3.31.

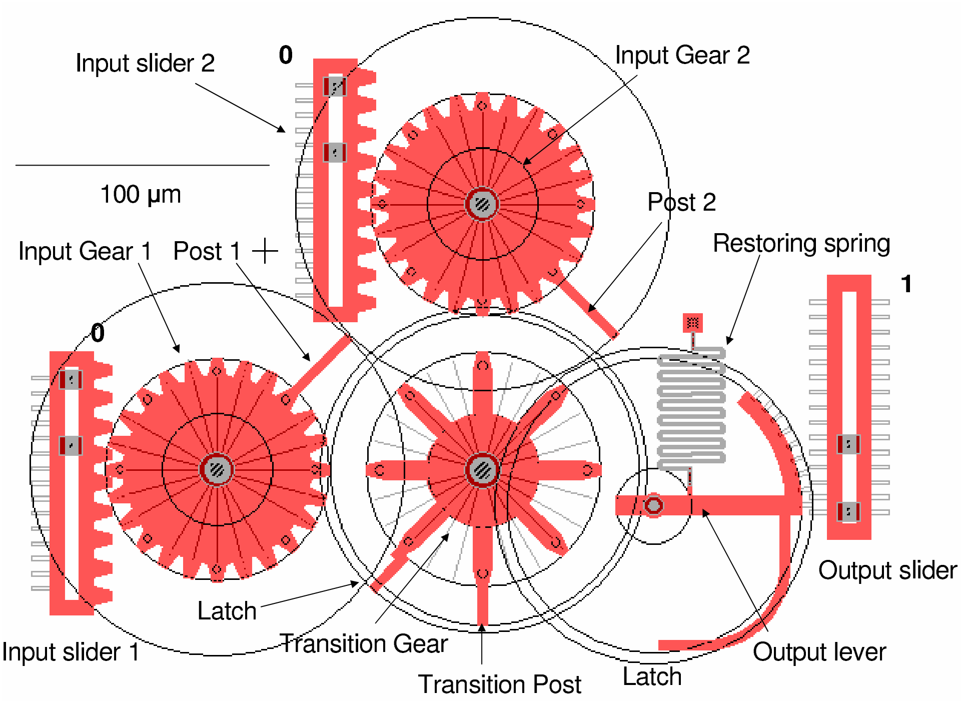


Figure 3.25 Computer aided drawing of the NAND-00 gate. NAND-1 structure with both inputs initially set to 0.

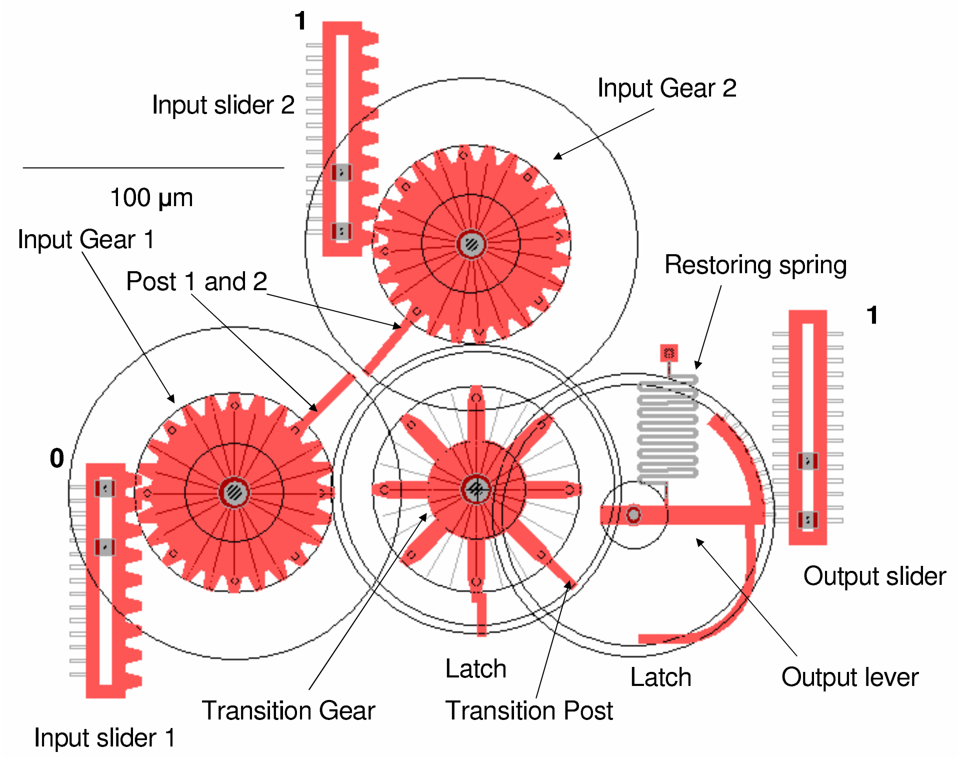


Figure 3.26 Computer aided drawing of the NAND-01 gate. NAND-1 structure with input 1 initially set to 0 and input 2 set to 1.

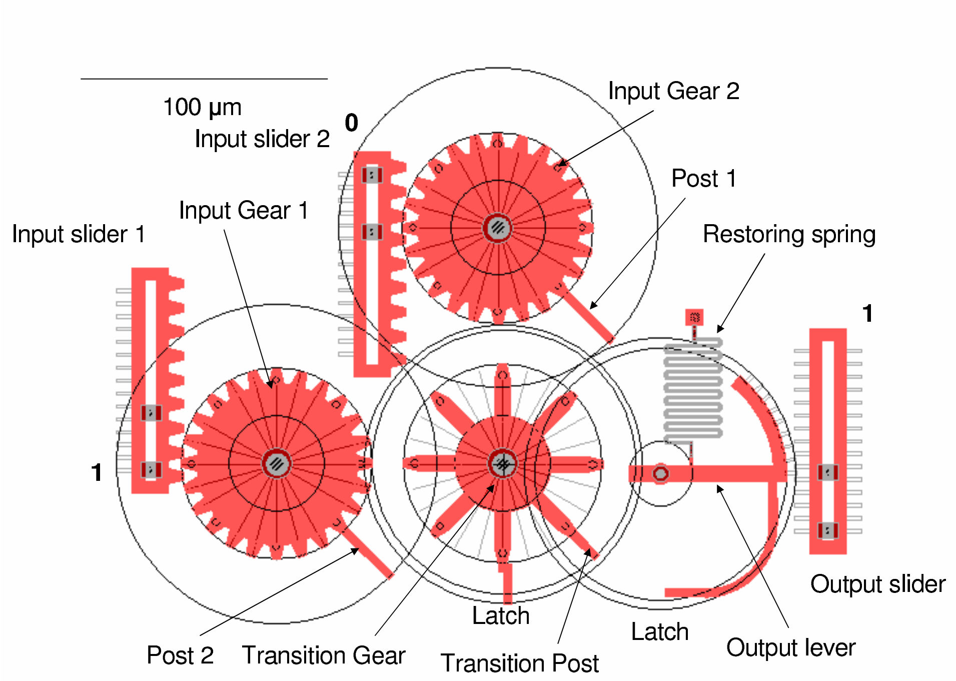


Figure 3.27 Computer aided drawing of the NAND-10 gate. NAND-1 structure with input 1 initially set to 1 and input 2 set to 0.

Before discussing the design of the NAND-0 structure, the design and operation of the NAND-1 structure is presented.

The following is an explanation of the operating principle behind the mechanical NAND-1 gate. This analysis is for the NAND gate with both inputs initially set to 0 (NAND-00), but a similar analysis can be applied to the other two NAND-1 gates. The terms used in this analysis refer to labels of NAND gate parts found in Figure 3.25, and PolyMUMPs process layers as discussed in Section 2.8.

For the NAND-00 gate (Figure 3.28), moving input slider 1 into the binary 1 position causes input gear 1 to rotate clockwise (CW) 90°. In this process, post 1 pushes against a gear tooth on the transition gear, causing it to rotate counterclockwise (CCW) 45°. This results in the transition post advancing 45° in the direction of the output lever. This has no impact on the output lever, the output slider, or input gear 2. The device is now in the 101 position (i.e., input 1 is in the binary 1 state, input 2 is in the binary 0 state, and the output remains in its 1 state), as

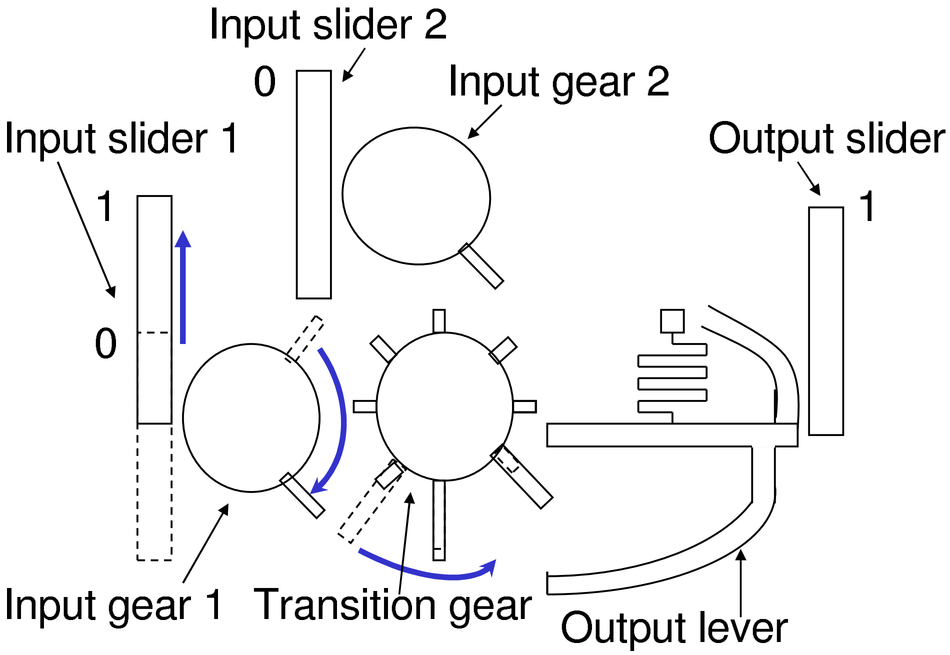


Figure 3.28 Simplified drawing of the NAND-00 gate. The blue arrows show the result of moving input slider 1 to the binary 1 position.

depicted by the full objects in Figure 3.28 and the ghosted objects in Figure 3.29. By moving input slider 2 to the binary 1 position, input gear 2 rotates 90° CW, post

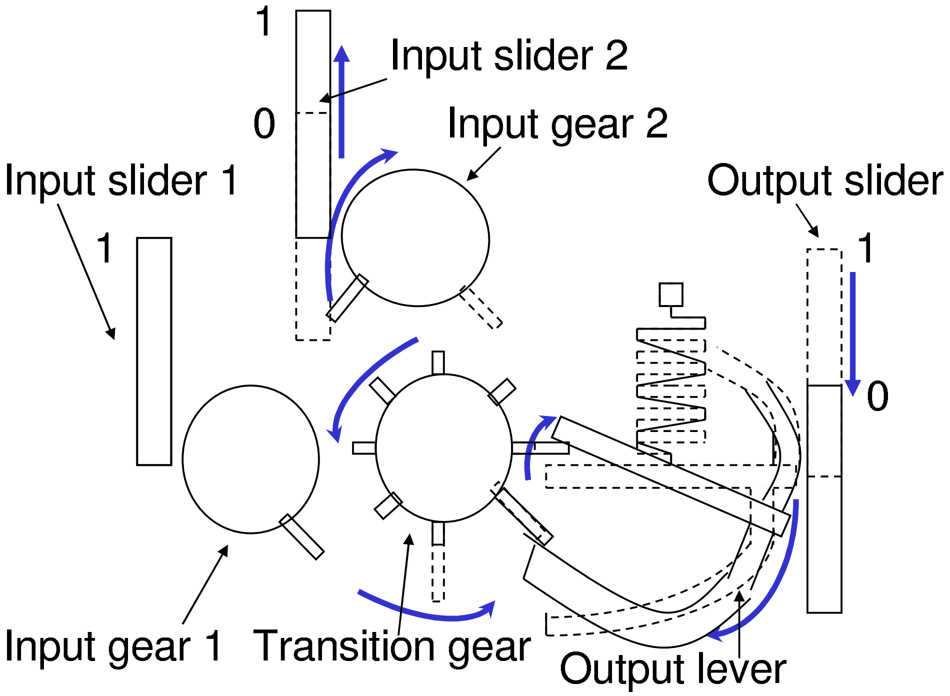


Figure 3.29 Simplified drawing of the NAND-00 gate after being moved to the 10 position. The blue arrows show the result of moving input slider 2 to the binary 1 position.

2 pushes against a gear tooth on the transition gear, which causes it to rotate an additional 45° CCW, and the transition post pushes against the end of the output lever. This causes the restoring spring to extend, and the teeth on the right side of the output lever to slide the output slider to the binary 0 position, as shown in Figure 3.29. Consequently, the poly1 latch on the lever will lock with the latch on the transition gear, causing the device to remain in this state until a CW force is applied to the transition gear. A simplified depiction of the locking mechanism is shown in Figure 3.30. Moving either of the two input sliders back to the binary 0 position will result in the necessary CW force. The post on the input gear corresponding to the input slider that is moved to the binary 0 position will push against a gear tooth on the transition gear, moving the transition gear 45° CW. The latches

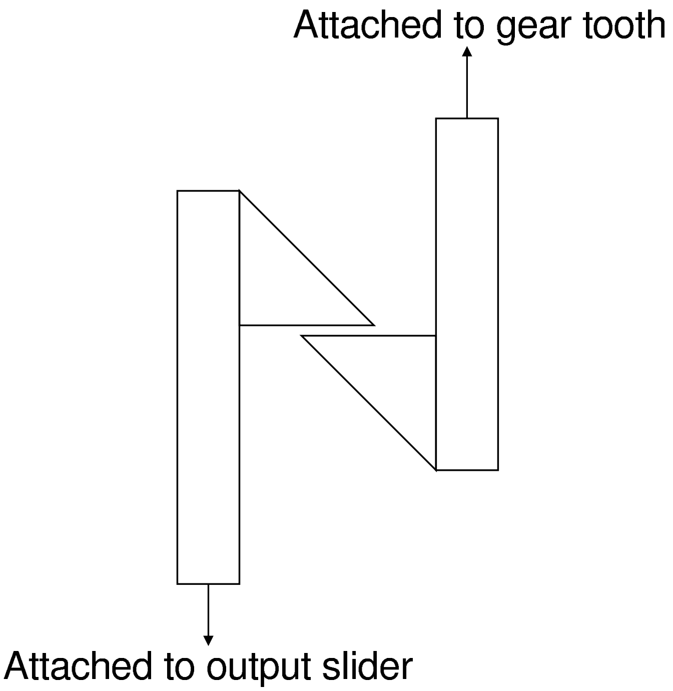


Figure 3.30 Simplified drawing of the locking mechanism for the NAND-1 gate.

are unlocked, the output lever is freed from the transition post, and the restoring spring contracts, causing the output lever to return to its original position and move the output slider back to the binary 1 position. This results in either the 101 or 011 state, depending on which gear is moved to the binary 0 position. By moving the other gear to the binary 0 position, the transition post is moved an additional 45 ° CW, and the device is restored to the 001 state.

The NAND-1 gate structure has two limitations. First, the inputs can not be applied simultaneously. The order in which the inputs are set is not important, but attempting to move the inputs at the same time will result in one of two scenarios. The first scenario is that if the inputs are being applied in such a way that the input gears are acting on the transition gear in opposite directions, they will get stuck and not be allowed to continue movement. The second scenario is that if the inputs are applied such that the input gears move the transition gear in the same direction, the total rotation of the transition gear subsequent to applying the simultaneous inputs will only be half of the rotation achieved by applying the inputs consecutively. The second limitation is that the fabrication process used for this device requires a minimum separation of 2 μm between structures. This results in a time delay between stages of a device, as well as unavoidable slop in the overall structure. In spite of these limitations, the NAND-1 gates should operate as designed.

Before discussing the differences between the NAND-0 gate structure (Figure 3.31) and the NAND-1 gate structure, a similar operational analysis is presented. For the NAND-11 gate, moving input slider 1 into the binary 0 position causes input gear 1 to rotate CCW 90°. In this process, post 1 pushes against a gear tooth on the transition gear, causing it to rotate counterclockwise (CCW) 45°. This results in the transition post advancing 45° CW, and causes the transition gear restoring spring to extend. In its 45° movement, the transition post pushes against the end of the output lever. This causes the restoring spring on the output lever to extend, and the

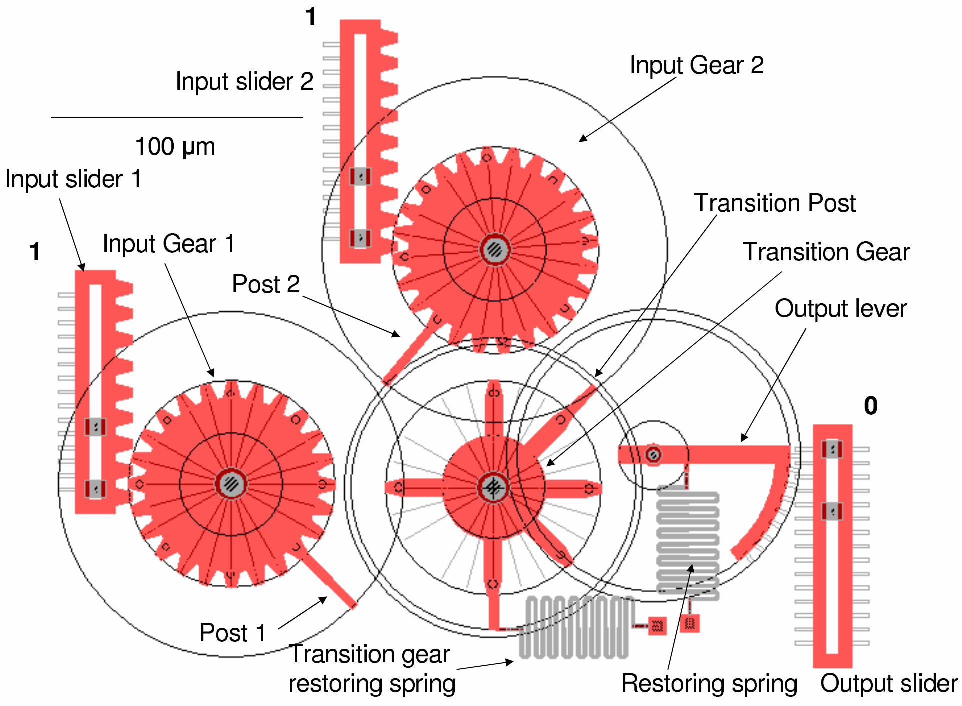


Figure 3.31 Computer aided drawing of the NAND-11 gate. NAND-0 structure with both inputs initially set to 1.

teeth on the right side of the output lever to slide the output slider to the binary 1 position. In order for the device to remain in this state, input slider 1 must be held in the binary 0 position, and post 1 must not be allowed to clear the transition gear tooth. With the device in the 011 state, input slider 2 can be moved to the binary 0 position without having any effect on the transition gear or the output, since as gear 2 is rotated CCW, post 2 is allowed to move freely to its final position, resting against the transition gear tooth next to the transition post. The device is now in the 001 position, and either of the input sliders can be moved back to the binary 1 position without affecting the output state. To clarify this point further, assume input 1 is moved to the binary 1 position. This results in input gear 1 rotating 90° CW, causing post 1 to no longer hold the transition gear in place. That is acceptable, since post 2 is in position to hold the transition gear in place. Until both sliders are moved to the binary 1 position, the output slider remains in the binary 1 state. With input slider 1 in the binary 1 state, moving input slider 2 to the binary 1 state causes post 2 to move 90° CW, away from the transition gear tooth. This allows the transition gear restoring spring to contract, causing the transition gear to move 45° CCW to its initial position. Consequently, the output lever is released, the output lever's restoring spring contracts, and the output slider is moved to the binary 0 state. The device is once again in its 110 state.

Similar to the NAND-1 gate structure, the NAND-0 structure also has two limitations. First, unlike the NAND-1 gates, the NAND-0 gate requires the inputs to be held in place in order for the output to remain in the binary 1 state. However, both inputs can be applied simultaneously without affecting proper device operation. The second limitation is (as before) that the fabrication process used for this device requires a minimum separation of 2 μm between structures, resulting in a time delay between stages of the device, as well as unavoidable slop in the overall structure. In spite of these limitations, the NAND-0 gate should operate as designed.

3.2.3 NOR gates. The NOR gate is a digital computing device that operates according to the truth table found in Table 3.3. This design was based on Kladitis' NOR gate from [2], and is an attempt to improve upon it. As with the NAND gates, sliders are used at the inputs and output, gears are used in place of levers, and all four physical gate configurations are presented. Figures 3.32, 3.33, 3.34, and 3.35 show the NOR gate designs for this thesis.

Input 1	Input 2	Output
0	0	1
0	1	0
1	0	0
1	1	0

Table 3.3 NOR truth table.

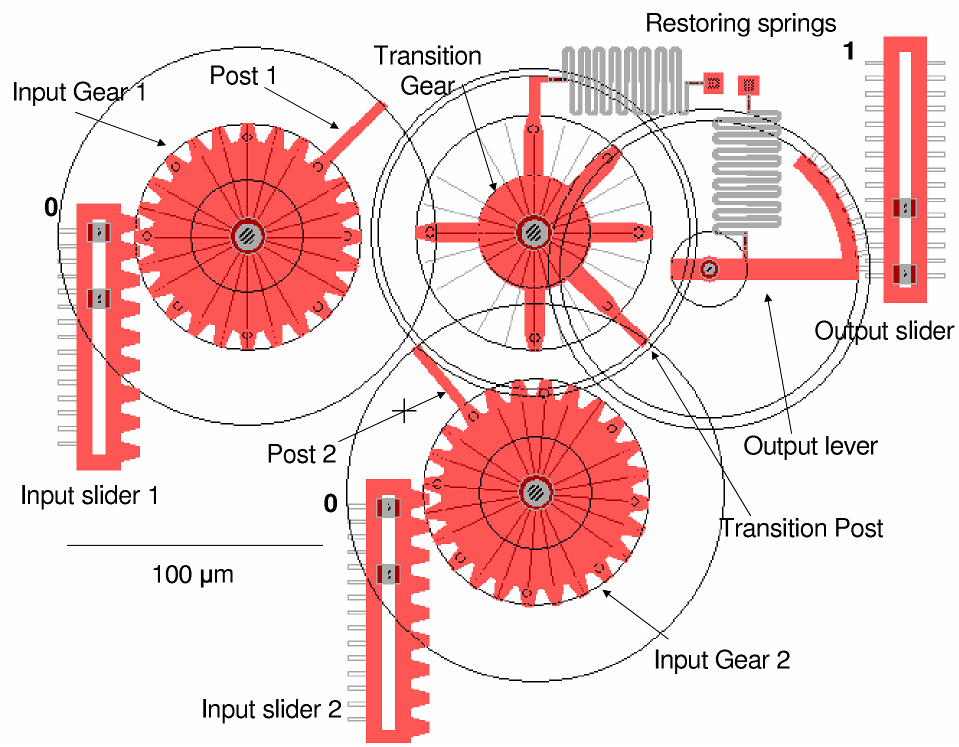


Figure 3.32 Computer aided drawing of the NOR-00 gate. NOR-1 structure with both inputs initially set to 0.

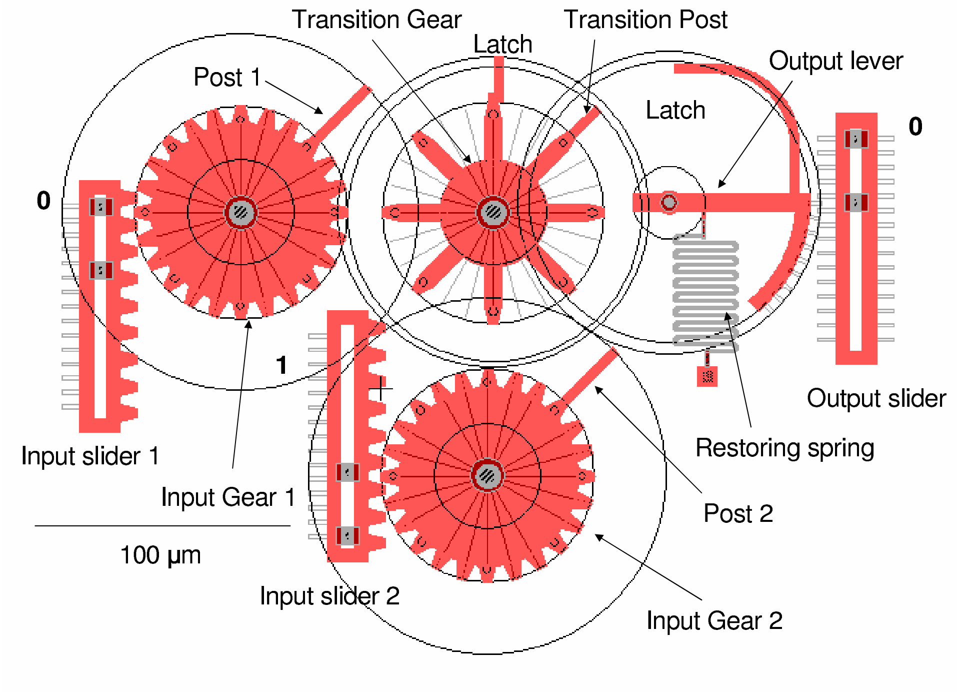


Figure 3.33 Computer aided drawing of the NOR-01 gate. NOR-0 structure with input 1 initially set to 0 and input 2 set to 1.

By comparing these figures with those presented for the NAND gates, it can be shown that a purely mechanical NOR gate is equivalent to a mechanically inverted or flipped NAND gate. Another way to view this would be by inverting every value in the truth table for the NAND gate (Table 3.2). The resulting table contains the four rows from the truth table for the NOR gate (Table 3.3). Since flipping the NAND structure upside down results in an inverse in the corresponding values, it is apparent that the new structure is a NOR gate. The design of NOR gates was achieved by creating the four NAND gate designs and physically flipping them upside down. One major benefit to this discovery is that by utilizing the exact same structure for both NOR and NAND gates, the same mechanical models can be used, and speed and force characteristics are identical for both. This will be applied in Chapter IV, where the logic gates are modelled to determine potential switching speeds.

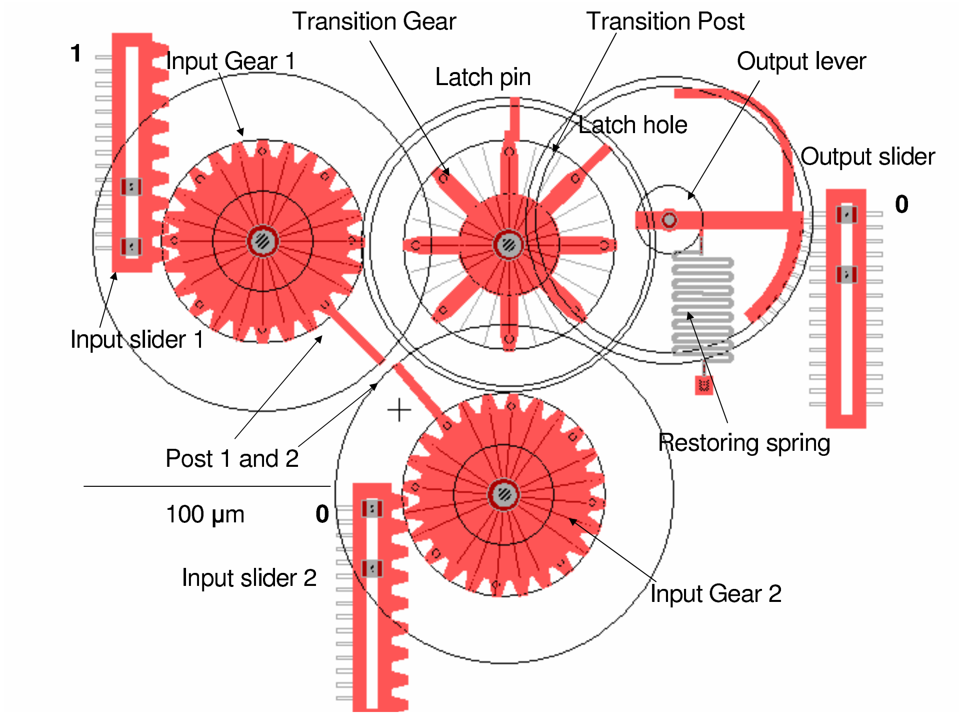


Figure 3.34 Computer aided drawing of the NOR-10 gate. NOR-0 structure with input 1 initially set to 1 and input 2 set to 0.

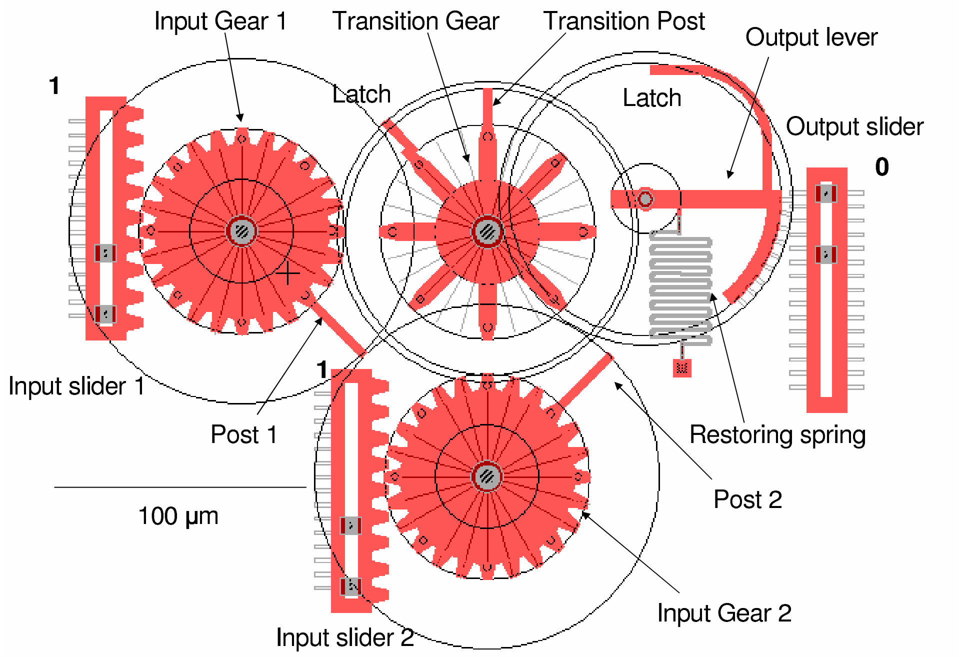


Figure 3.35 Computer aided drawing of the NOR-11 gate. NOR-0 structure with both inputs initially set to 1.

3.2.4 XOR gates. All digital logic can be accomplished with a combination of NANDs, NORs, and inverters. The design of an XOR gate is an attempt to accomplish this mechanically. The XOR gate is a digital logic operator that operates in accordance with the truth table found in Table 3.4. Using boolean algebra, the XOR operation can be performed using only two inverters and three NAND gates. Following is the derivation of the XOR operation showing how this can be accomplished:

$$A \bigoplus B = \{A \cdot \bar{B}\} + \{B \cdot \bar{A}\} = \overline{\overline{A \cdot \bar{B}} \cdot \overline{B \cdot \bar{A}}} \tag{3.13}$$

Input 1	Input 2	Output
0	0	0
0	1	1
1	0	1
1	1	0

Table 3.4 XOR truth table.

This design was based on Kladitis' XOR gate from [2], and is an attempt to improve upon it. The two main improvements that were made are that there are only two, rather than four, inputs, and that all four combinations of inputs (00,01,10,and 11) have been created. These are found in Figures 3.36, 3.37, 3.38, and 3.39, respectively.

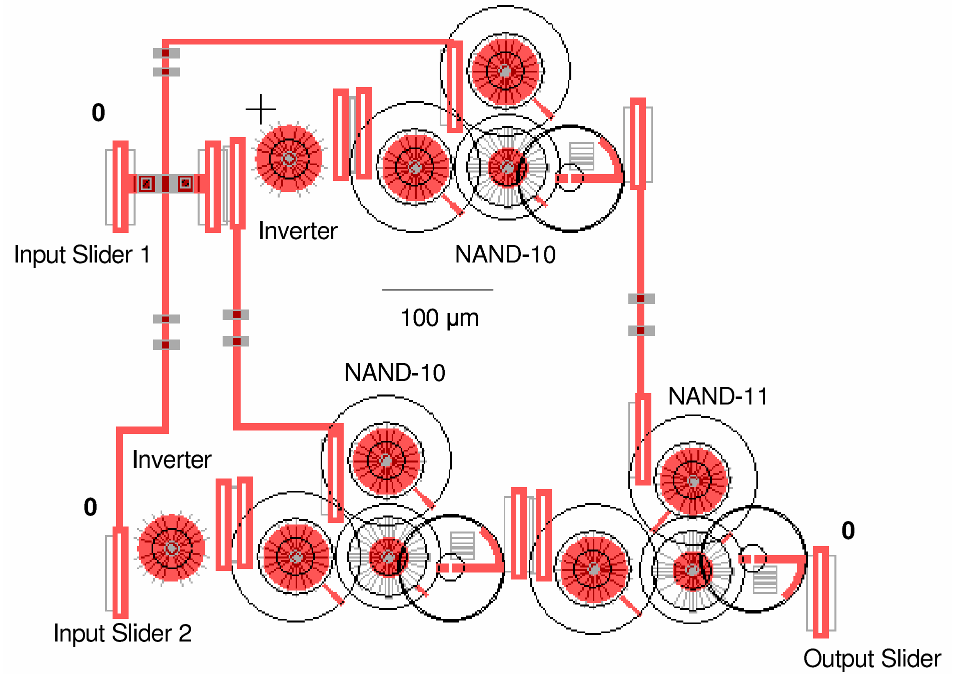


Figure 3.36 Computer aided drawing of the XOR-00 gate. XOR gate with both inputs initially set to 0.

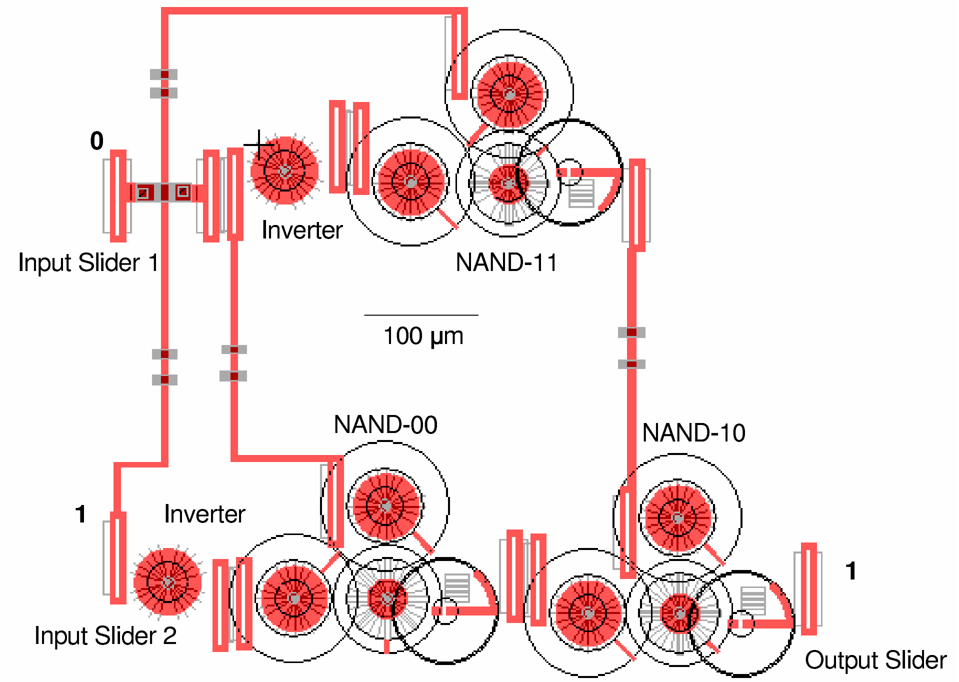


Figure 3.37 Computer aided drawing of the XOR-01 gate. XOR gate with input 1 initially set to 0 and input 2 set to 1.

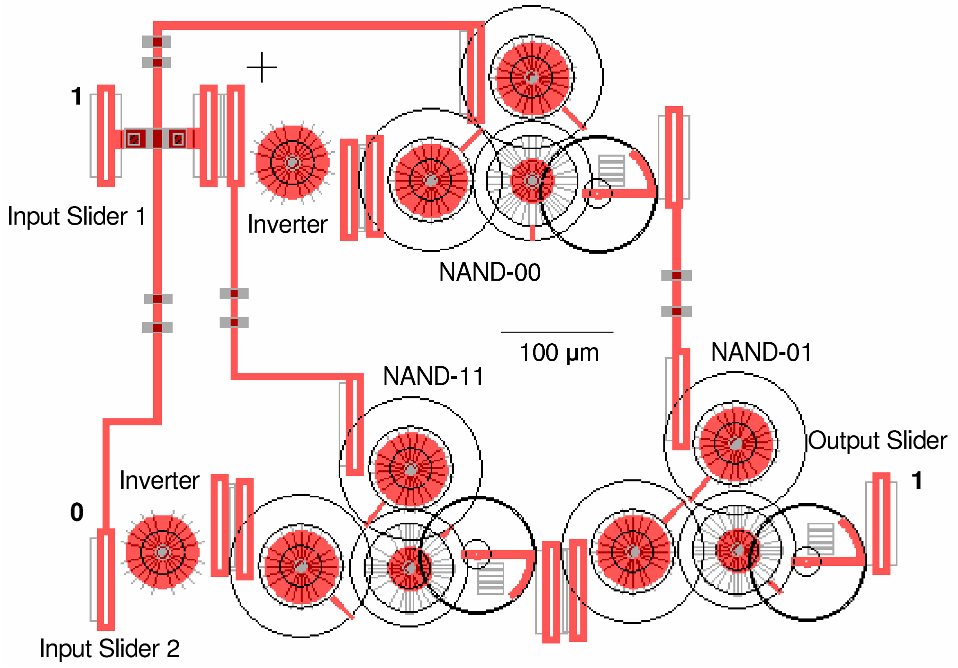


Figure 3.38 Computer aided drawing of the XOR-10 gate. XOR gate with input 1 initially set to 1 and input 2 set to 0.

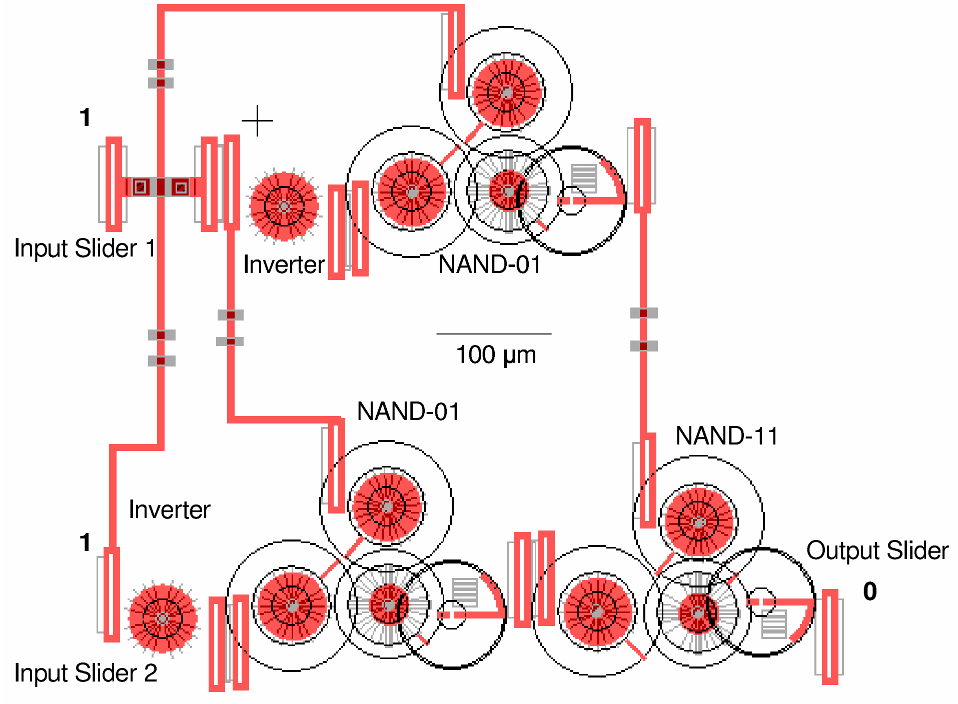


Figure 3.39 Computer aided drawing of the XOR-11 gate. XOR gate with both inputs initially set to 1.

3.3 Converters

In some cases, it is necessary to convert a sequence of digital inputs into an analog output or an analog input into a sequence of digital outputs. For mechanical computing devices, this is accomplished with mechanical digital-to-analog (D-to-A) and analog-to-digital (A-to-D) converters. This section presents the inspiration for these designs, including an explanation of the device operation. The D-to-A converter is presented first, since this device has already been fabricated and tested. The A-to-D converter has been designed, but is in the process of being fabricated at this time. Testing will be performed as future work.

3.3.1 Digital-to-analog (D-to-A) converters. MEMS digital-to-analog (D-to-A) converters are needed to convert a series of mechanical digital inputs to mechanical analog outputs. The design for this research was inspired by Yeh's micro-electromechanical digital-to-analog converter (MEMDAC) [6], which uses a cascade of lever arms to represent bits. The D-to-A converter that was designed for this thesis has an input slider at each of the lever arms that can be connected to the output sliders of a digital logic gate. Figure 3.40 shows a 2-bit mechanical D-to-A converter. Both input sliders are initially set to the binary 0 position. The least significant bit (LSB) is on the left, and the most significant bit (MSB) is on the right. A simplified model of the MEMDAC is shown in Figure 3.41. The insert of Figure 3.41(a) portrays the gap between the lever arm and bumper. Each slider is designed to move the gap distance, g, of 6 μm when a binary 1 input is applied to the given bit. Setting the LSB to a binary 1 while keeping the MSB at 0 results in a shift of $\frac{g}{4}$ or 1.5 μm in the analog output, as shown in Figure 3.41(b). Setting the MSB to 1 with the LSB at 0 results in a shift of $\frac{g}{2}$ or 3 μm, as shown in Figure 3.41(c). Setting both values to 1 results in a combined shift of $\frac{3g}{4}$ or 4.5 μm, as shown in Figure 3.41(d).

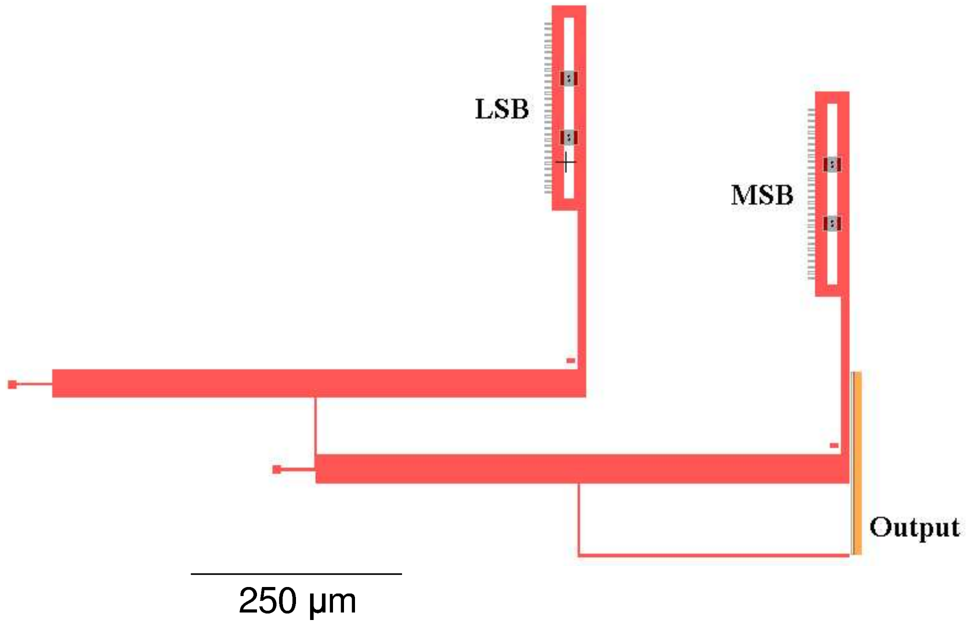


Figure 3.40 Computer aided drawing of the 2-bit mechanical digital-to-analog converter.

For a given N-bit D-to-A converter, the analog output is equal to the average of the displacement of the MSB (Bit$_{N-1}$) and the displacement of the output to the previous stage (Output$_{N-1}$). Similarly, Output$_{N-1}$ is equal to the average of the displacement of Bit$_{N-2}$ and Output$_{N-2}$. With the displacement of a given bit equal to the product of the gap distance g and the binary value assigned to the bit, B_i (1

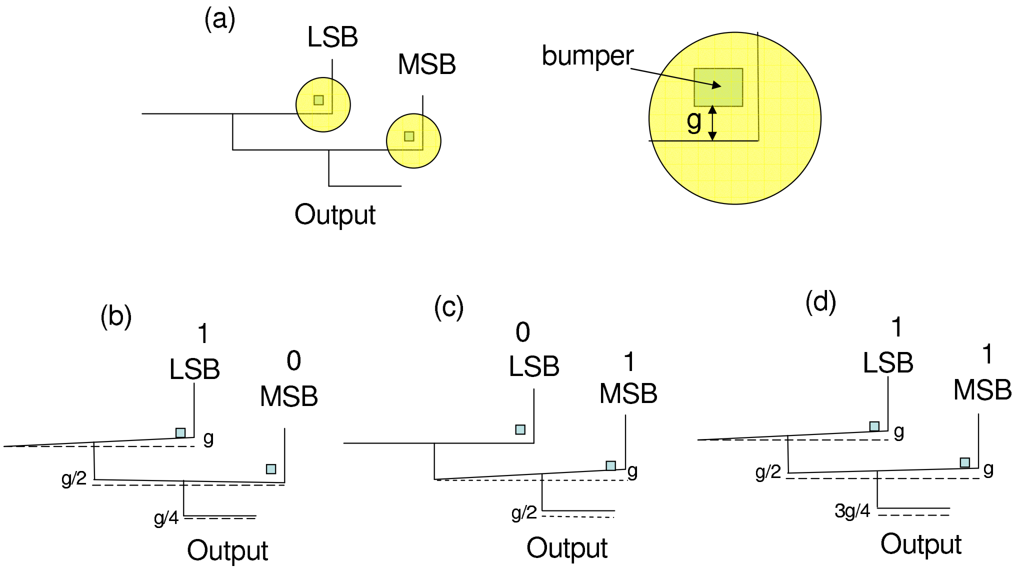


Figure 3.41 Simplified model of a 2-bit MEMDAC. (a) Both inputs set to 0, as fabricated, with the inset portraying the gap and bumper. (b) Setting LSB to 1 and MSB to 0 results in an output of $\frac{g}{4}$. (c) Setting MSB to 1 and LSB to 0 results in an output of $\frac{g}{2}$. (d) Setting both MSB and LSB to 1 results in an output of $\frac{3g}{4}$.

or 0), the analog output of an N-bit digital-to-analog converter can be written as [6]

$$\begin{aligned} Output &= \frac{gB_{N-1} + Output_{N-1}}{2} \\ &= \frac{gB_{N-1} + \frac{gB_{N-2} + Output_{N-2}}{2}}{2} \\ &= \frac{gB_{N-1} + \frac{gB_{N-2} + \frac{gB_{N-3} + Output_{N-3}}{2}}{2}}{2} \\ &= \cdots \\ &= g \sum \frac{B_i}{2^{N-i}} \\ &= g \sum \frac{2^i}{2^N} B_i \qquad (3.14) \end{aligned}$$

where g is the gap distance, N is the number of bits, and B_i represents the value of the i^{th} bit, 1 or 0.

3.3.2 Analog-to-digital (A-to-D) converters. A purely mechanical analog-to-digital (A-to-D) converter is used to convert a continuous mechanical input into a discrete (digitized) mechanical output. Upon examining the relationship between analog and digital movements, gears appeared to be the ideal structure for converting the movement of a linear analog slider into a set of binary outputs. The gear train idea comes from the fact that a gear of radius R will make two full revolutions in the time that it takes a gear next to it with radius 2R to complete one full revolution. Similarly, a gear with radius 4R rotates once in the time it takes the gear with radius 2R to rotate twice, and the gear with radius R to rotate four times. By drawing a line through the center of the gears, and placing a gold dot on the bottom half of each of the gears, the output of each gear (or bit) is said to be 0. A binary 0 corresponds to the gold spot for each gear located under the line, and a binary 1 corresponds to the gold spot being on top of the line. Figure 3.42 shows a simplified drawing of this 3-bit A-to-D converter. By moving the input slider, the corresponding input and output for a 3-bit A-to-D converter is shown in Table 3.5. With Gear R representing the least significant bit (LSB), and Gear 4R representing the most significant bit (MSB), a 4-bit opto-mechanical A-to-D converter has been designed for fabrication

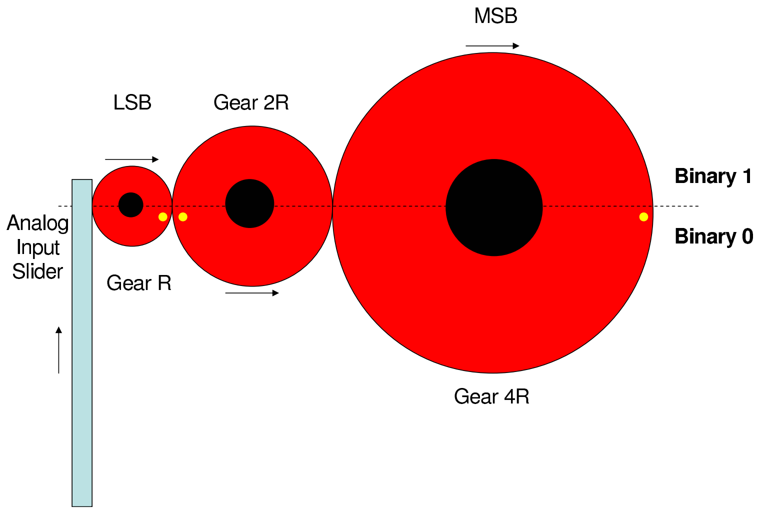


Figure 3.42 Simplified depiction of a 3-bit Opto-mechanical analog-to-digital (A-to-D) converter.

Analog Input	Gear R	Gear 2R	Gear 4R
0	0	0	0
π R	1	0	0
2π R	0	1	0
3π R	1	1	0
4π R	0	0	1
5π R	1	0	1
6π R	0	1	1
7π R	1	1	1

Table 3.5 A-to-D converter input-output relationship

in the PolyMUMPs process, where the output can be read by an optical reader that senses light reflecting from the gold dot only if the dot is in the binary 1 position. A picture of this design is shown in Figure 3.43.

Gold dots are fabricated on the gears in Figure 3.43 similar to those of Figure 3.42. This device requires a series of optical sensors to read the output, and is not compatible with the other purely mechanical devices for this research. A purely mechanical output is desired for the A-to-D converter. After further brainstorming, and a discussion with a fellow colleague [35], the purely mechanical A-to-D converter was designed. Due to the fabrication restrictions of the PolyMUMPs process, which, among other things, limits the designer to two movable structural layers, an extra gear was used as a spacer between output gears to prevent the mechanical cams from obstructing one another during operation. Figure 3.44 shows the design for the purely mechanical A-to-D converter. The device operates as follows. The analog input slider on the left causes the gears to move. Each of the output gears moves CCW. The poly2 cam structures are fabricated on a poly1 platform to elevate them above the poly1 gears. This allows them to move across the top of the poly1 gears with minimal surface rubbing. As the poly2 cam makes contact with the output lever arm, the lever arm is pushed up, and the restoring spring is extended. This position represents a binary 1. Once the cam has rotated through, the restoring

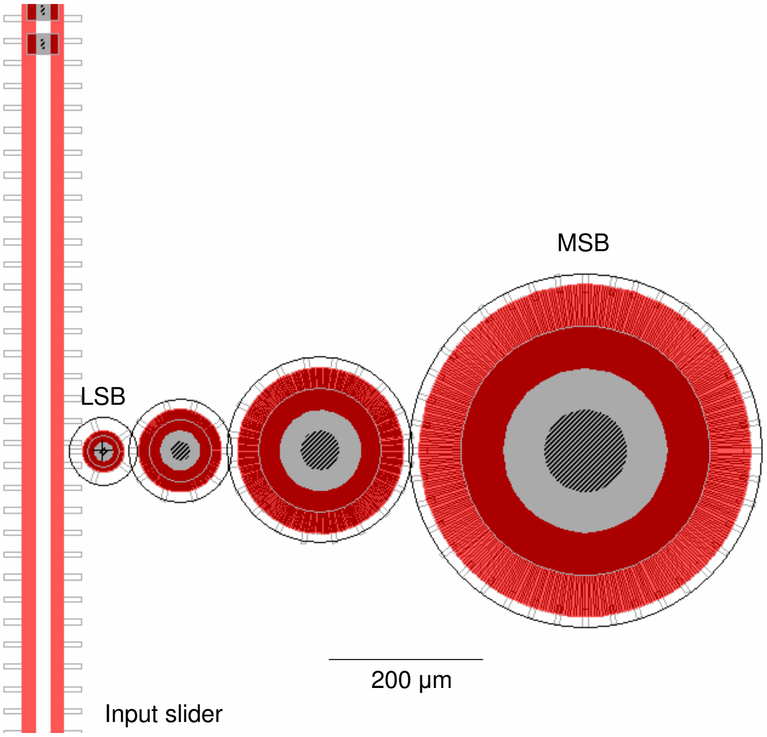


Figure 3.43 Computer aided drawing of a 4-bit opto-mechanical analog-to-digital (A-to-D) converter.

spring returns the output lever arm to its initial binary 0 state. Table 3.5 represents the mechanical input-output relationship of the A-to-D converter of Figure 3.44.

This concludes the discussion of the purely mechanical MEMS computing designs. Chapter IV presents models of the digital computing devices in order to characterize them in terms of potential switching speed.

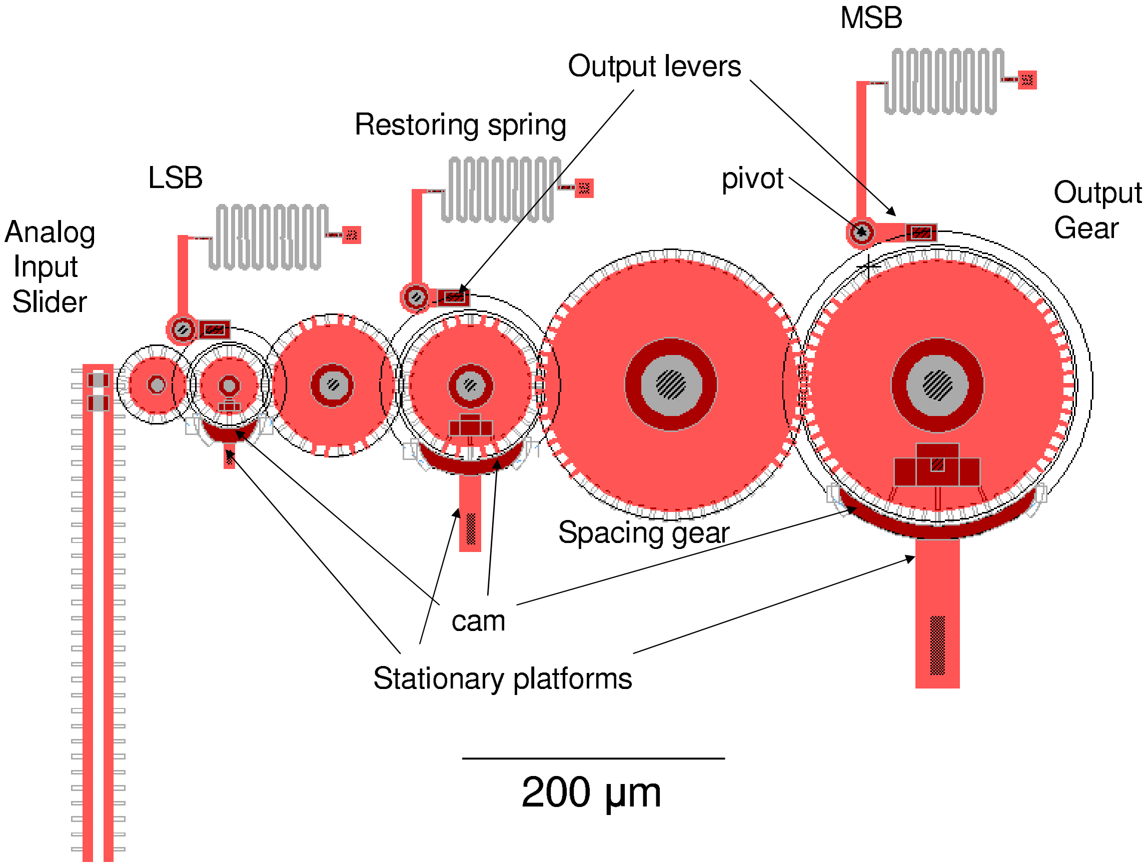


Figure 3.44 Computer aided drawing of the purely mechanical analog-to-digital (A-to-D) converter.

IV. Mechanical Modelling

One of the most important characteristics of a computing device, other than whether it performs the correct operation, is how fast the operation is performed. In electronic computer systems, this is referred to as switching speed. The same term will be used with respect to the mechanical computing devices for this thesis.

In order to determine the potential switching speed of the mechanical computing devices, each device is separated into its most basic elements. Models for each of these elements is presented. Using the elemental models as a basis set, models of the actual computing devices are derived. For all models, it is assumed that the devices have been fabricated using the PolyMUMPs surface micromachining process, and that they are operating in air at room temperature at a pressure of 1 atm.

4.1 Elemental Models for Mechanical Computing Devices

Each of the digital computing designs can be broken down into four basic elements. These are springs, sliders, gears, and output levers. Determining the switching speed of a given mechanical computing device requires a knowledge of how fast each mechanical element of the system can move, and how they interact with one another. This section focuses on the former, and Section 4.2 focuses on the latter. For a given element, a simplified mass-spring-damper system equation is set equal to a maximum input force, since the maximum velocity of a system is limited by its input force. A survey of MEMS literature was conducted with the goal of finding a high-force MEMS actuator that would be compatible with the movement of these devices. A four-plate scratch drive actuator (SDA) has been theoretically shown to produce a force of 850 μN [36]. This value of input force will be used as a constant input force acting on each element. All models assume that initial displacement and velocity are both 0. The first element that will be modelled is the spring, followed by the slider, gears, and output lever.

4.1.1 Spring restoring force. The basic structure of a spring, as shown in Figure 4.1, is that of a beam that is fixed at one end and free to move at the opposite end. The elastic compliance of such a beam is needed in order to determine the compliance of the spring. The compliance of the beam with one fixed end is [3]

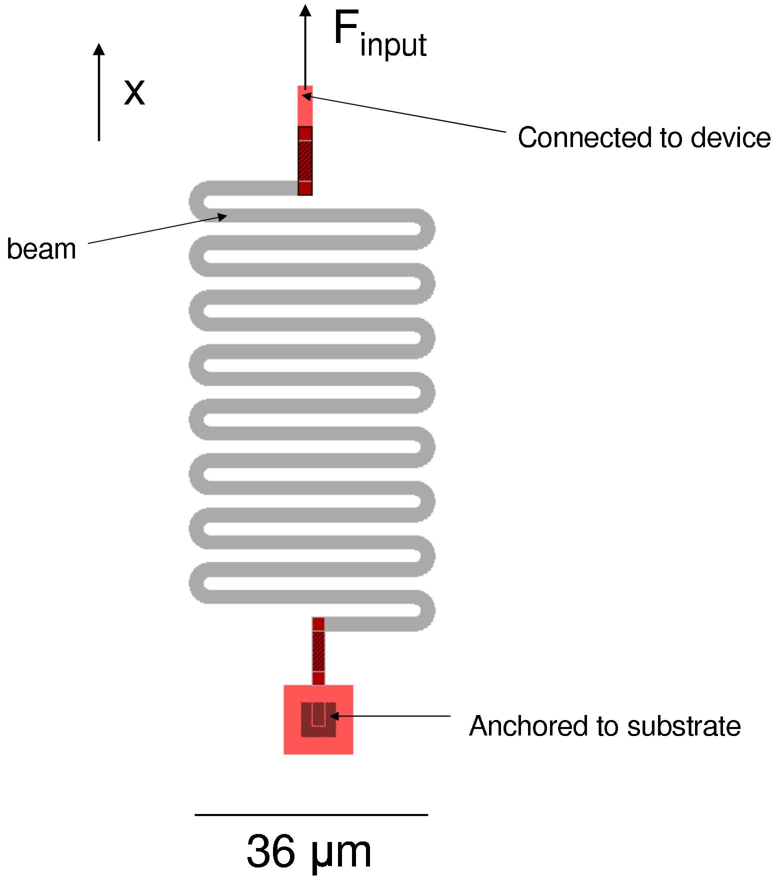


Figure 4.1 Computer aided drawing of the spring element. This is a poly2 spring anchored to the substrate at one end and connected to a device at the other end. It consists of 15 bars of length 36 μm. The height of the bars is 1.5 μm and the width is 2 μm.

$$C_{beam} = \frac{L^3}{12EI} \quad \left[\frac{\mathrm{m}}{\mathrm{N}}\right] \tag{4.1}$$

where L is the length of the beam in meters (m), E is Young's modulus for the beam material in Pascals (Pa), and I is the moment of inertia of the beam in m^4. The moment of inertia of the beam with height h and width w is [3]

$$I_{beam} = \frac{hw^3}{12} \ \left[\mathrm{m}^4\right] \tag{4.2}$$

By inserting Equation 4.2 into Equation 4.1, the compliance of the beam can be reduced to purely geometrical and material parameters.

$$\begin{aligned} C_{beam} &= \frac{L^3}{12E\frac{hw^3}{12}} \\ &= \frac{1}{Eh}\left(\frac{L}{w}\right)^3 \ \left[\frac{\mathrm{m}}{\mathrm{N}}\right] \end{aligned} \tag{4.3}$$

The compliance of the spring structure, C_{spring}, depends on the number of beams used to construct the spring,

$$C_{spring} = NC_{beam} \ \left[\frac{\mathrm{m}}{\mathrm{N}}\right] \tag{4.4}$$

where N is the number of beam elements. The spring constant for the spring is equal to the inverse of its compliance.

$$\begin{aligned} k_{spring} &= \frac{1}{C_{spring}} \\ &= \frac{1}{NC_{beam}} \\ &= \frac{1}{N\frac{1}{Eh}\left(\frac{L}{w}\right)^3} \\ &= \frac{Eh}{N}\left(\frac{w}{L}\right)^3 \ \left[\frac{\mathrm{N}}{\mathrm{m}}\right] \end{aligned} \tag{4.5}$$

The parameters used for the spring element are:

$E = 160$ GPa (polysilicon spring) [37]

$h = 1.5\ \mu$m (poly2 layer thickness)

$N = 15$

$w = 2\ \mu$m (minimum width allowed in PolyMUMPs process)

$L = 36\ \mu$m

This gives a spring constant of $k_{spring} = 2.74\ \frac{N}{m}$. With this parameter, the spring can be modelled with respect to force.

$$F_{spring} = k_{spring}x \text{ [N]} \tag{4.6}$$

where x is the displacement of the spring. Applying a force of 850 μN would theoretically cause the spring to stretch 310 μm. With the analysis of the spring complete, the slider will now be modelled.

4.1.2 Slider dynamic response. The slider (see Figure 4.2) is a poly1 structure with dimples that make contact with the nitride layer on the bottom and poly2 guides that overlap the structure in four locations on the top. The slider exhibits translational motion, which can be characterized by simplifying the following mass-spring-damper equation.

$$mx'' + bx' + kx = F \tag{4.7}$$

where F is the net force acting on the slider, m is the mass of the slider, x'' is the slider's acceleration, b is the damping constant of the slider, x' is the slider's velocity, k is the spring constant of the slider, and x is the slider's displacement. By making appropriate assumptions, Equation 4.7 can be simplified further. This model assumes that $k = 0$, since there is no spring in the slider element. It also assumes that there are friction forces, other than damping, that must be accounted

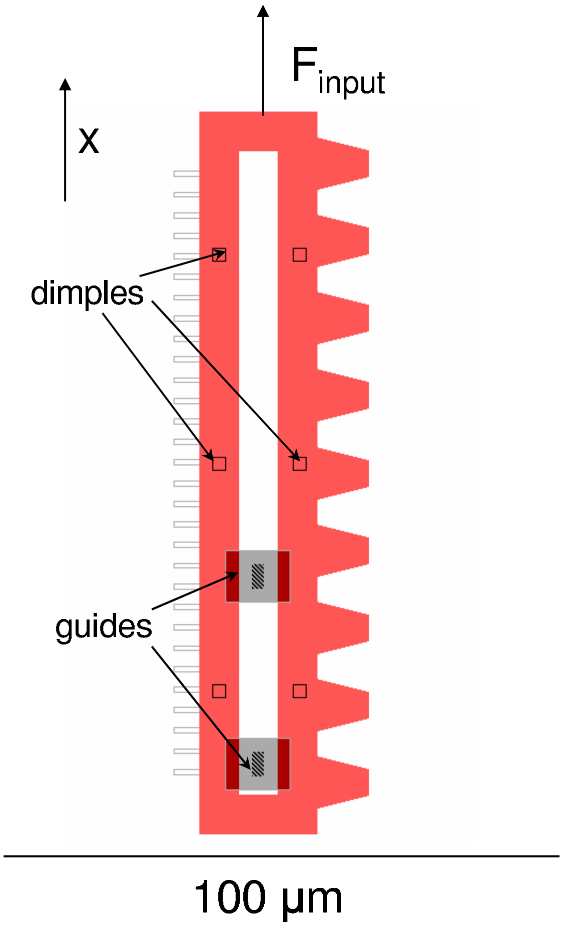


Figure 4.2 Computer aided drawing of the slider element. This is a poly1 slider with dimples that make contact with the nitride surface and poly2 guides that make contact with the top surface.

for due to surface rubbing. Based on these assumptions, Equation 4.7 becomes

$$mx'' + bx' = F_{input} - F_{friction} \tag{4.8}$$

where $F_{friction}$ is the total friction force and F_{input} is the constant 850 μ N force applied by the SDA. Equation 4.8 can be rewritten in the simplified differential equation form as

$$x'' + \frac{b}{m}x' = \frac{F_{input} - F_{friction}}{m} \tag{4.9}$$

Solving Equation 4.9 for the displacement, x, gives the analytical solution

$$x(t) = \frac{F_{input} - F_{friction}}{b}\left(\frac{m}{b}e^{-\frac{b}{m}t} + t - \frac{m}{b}\right) \tag{4.10}$$

with the exponential term in the equation representing the time constant for the slider element,

$$\begin{aligned} \tau_{slider} &= \frac{m}{b} \\ &= 194\mu s \end{aligned} \tag{4.11}$$

Taking the first and second derivatives of Equation 4.10 results in analytical solutions for the velocity, x', and acceleration, x'', of the slider with respect to time.

$$x'(t) = \frac{F_{input} - F_{friction}}{b}\left(1 - e^{-\dfrac{b}{m}t}\right) \tag{4.12}$$

$$x''(t) = \frac{F_{input} - F_{friction}}{m}\left(e^{-\frac{b}{m}t}\right) \tag{4.13}$$

The damping coefficient, b, is calculated as [38]

$$b = \frac{\eta_{eff}A}{d} \quad \left[\frac{kg}{s \cdot m^2}\right] \tag{4.14}$$

where η_{eff} is the effective viscosity of air, A is the top surface area of the slider and d is the air gap between the bottom of the slider and the nitride layer, caused by the 0.75 μm dimples. The effective viscosity of air, η_{eff} is [38]

$$\eta_{eff} = \frac{\eta}{1 + 2K_n} \quad \left[\frac{\mathrm{kg}}{\mathrm{m} \cdot \mathrm{s}}\right] \tag{4.15}$$

where η is the dynamic viscosity of air and K_n is the Knudsen number. The Knudsen number is [39]

$$K_n = \frac{\eta\sqrt{2RT}}{Pd} \quad \text{[dimensionless]} \tag{4.16}$$

where R is the ideal gas law constant, T is the temperature, and P is the pressure. By inserting Equation 4.16 into Equation 4.15, and then inserting the resulting value for η_{eff} into Equation 4.14, the damping coefficient for the slider can be calculated. Equation 4.17 shows the solution for b, followed by a list of the parameters that were used to solve the equation, with references.

$$\begin{aligned}
b &= \frac{\frac{\eta A}{d}}{1 + 2\frac{\eta\sqrt{2RT}}{Pd}} \\
&= \frac{\frac{1.8\times10^{-5}\frac{\mathrm{kg}}{\mathrm{m\cdot s}}wL}{0.75\mu m}}{1 + 2\frac{1.8\times10^{-5}\frac{\mathrm{kg}}{\mathrm{m\cdot s}}\sqrt{2(8.314\frac{\mathrm{J}}{\mathrm{mol\cdot K}})(293\mathrm{K})}}{(101325\frac{\mathrm{N}}{\mathrm{m}^2})(0.75\mu\mathrm{m})}} \\
&= 0.0165wL\frac{\mathrm{kg}}{\mathrm{s\cdot m^2}}\sqrt{\frac{\mathrm{kg}}{\mathrm{mol}}} \\
&= 0.0165wL\frac{\mathrm{kg}}{\mathrm{s\cdot m^2}}\sqrt{\frac{\mathrm{kg}}{\mathrm{mol}}\frac{1\mathrm{mol_{air}}}{0.029\mathrm{kg}}} \\
&= 20wL\ \left[\frac{\mathrm{kg}}{\mathrm{s\cdot m^2}}\right]
\end{aligned} \tag{4.17}$$

where

η $= 1.8\times10^{-5}\ \frac{\mathrm{kg}}{\mathrm{m\cdot s}}$ [40]

R $= 8.314\ \frac{\mathrm{J}}{\mathrm{mol\cdot K}}$ [41]

T $= 293$ K (room temperature)

$P\ = 101325\ \frac{\mathrm{N}}{\mathrm{m}^2}$ (atmosphere)

$d\ = 0.75\ \mu$m (air gap due to dimples)

Therefore, the slider has a damping coefficient of $b = 6.3{\times}10^{-8}\ \frac{\mathrm{kg}}{\mathrm{s}}$. The mass of the slider, m, is calculated by multiplying the density of polysilicon, $\rho_{poly} = 2.33\ \frac{\mathrm{g}}{\mathrm{cm}^3}$ [42], by the volume of the slider, which is assumed to be a rectangular solid with dimensions of L, w, and h, which represent the length, width, and height of the poly1 slider, respectively. The slider element designed for this thesis is of width $w = 22.5\ \mu$m and length $L = 140\ \mu$m, with $h = 2\ \mu$m corresponding to the thickness of the poly1 layer in the PolyMUMPs process. The friction force, $F_{friction}$ of the slider is

$$F_{friction} = (\mu_{pp} + \mu_{pn})\rho_{poly} mg \tag{4.18}$$

where $\mu_{pp} = 0.54$ [43] and $\mu_{pn} = 0.30$ [44] are the coefficients of friction for polysilicon on polysilicon and polysilicon on silicon nitride, and g is the acceleration of gravity. Inserting appropriate values into Equation 4.18 results in a friction force of

$$\begin{aligned} F_{friction} &= (0.54 + 0.33)(2.33 \times 10^3 \frac{\mathrm{kg}}{\mathrm{m}^3})(140\mu m)(22.5\mu m)(2\mu m)(9.8\frac{\mathrm{m}}{\mathrm{s}^2}) \\ &= 1.25 \times 10^{-4} \mu N \end{aligned} \tag{4.19}$$

By setting F_{input} to 850 μN and inserting the values of b and $F_{friction}$ into Equations 4.10, 4.12, and 4.13, the time-dependent behavior of the slider can be calculated. Figure 4.3 shows displacement, velocity, and acceleration of the slider with respect to time. The maximum velocity of the slider is 96 $\frac{\mathrm{m}}{\mathrm{s}}$, and it takes 1.664 μs to move the entire distance of 80 μm.

4.1.3 Gear dynamic response. Unlike the slider, which exhibits translational motion, the gear's motion is rotational. This section contains two different models. The first is for a gear that is free to rotate without a restoring spring attached. The second is for a gear with a restoring spring. In both cases, the model

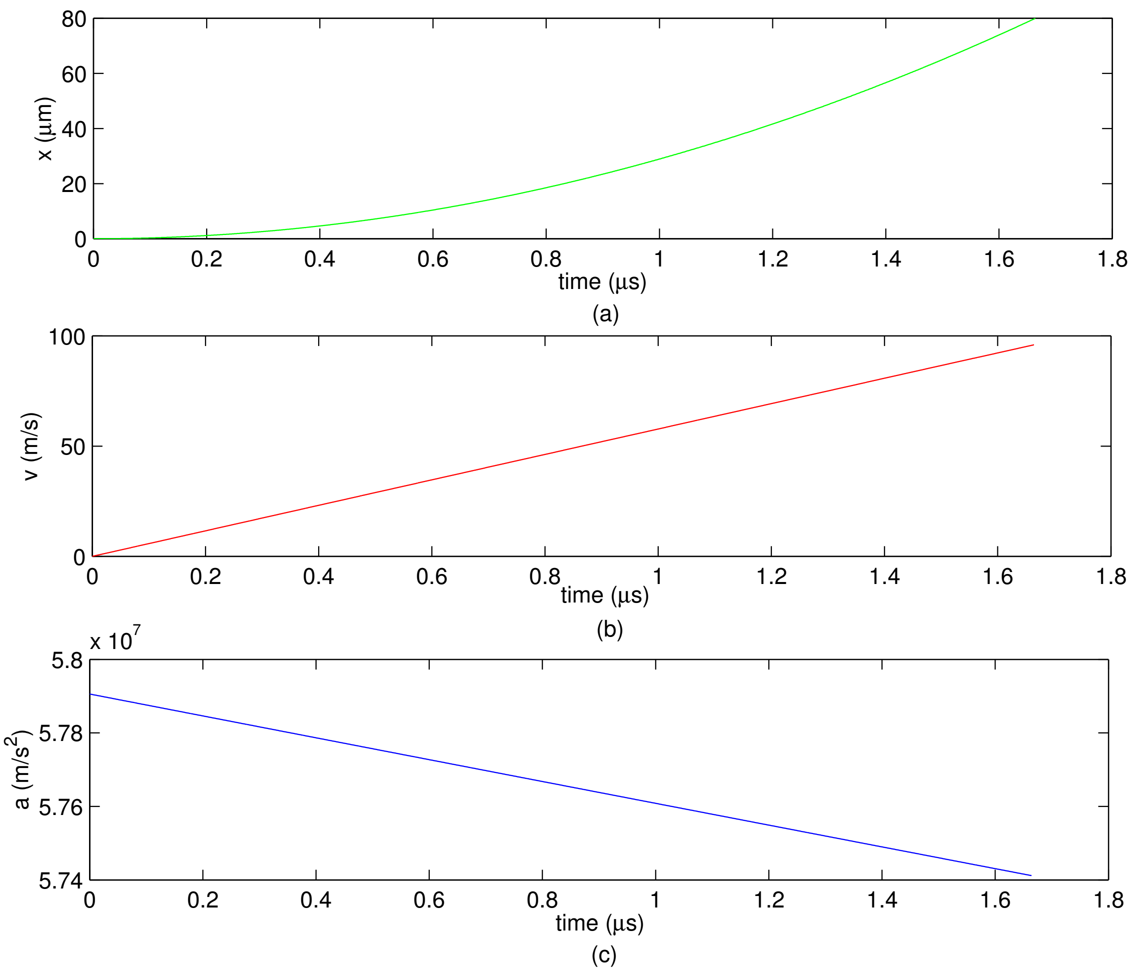


Figure 4.3 Plot of time-dependent behavior of the poly1 slider with a constant input force of 850 μ N. (a) Slider displacement vs. time. (b) Slider velocity vs. time. (c) Slider acceleration vs. time.

begins with the rotational mass-spring-damper system equation,

$$I\theta'' + c\theta' + G\theta = T \tag{4.20}$$

where I is the gear's moment of inertia, θ'' is the angular acceleration of the gear, c is the gear's damping coefficient, θ' is the angular velocity of the gear, G is the gear's spring constant, θ is the angular displacement of the gear, and T is the net torque applied to the gear. The model for the gear without a restoring spring is discussed first.

4.1.3.1 Gears without a restoring spring. For the input gear, which does not have a restoring spring (Figure 4.4), Equation 4.20 is reduced to

$$I\theta'' + c\theta' = T_{input} - T_{friction} \tag{4.21}$$

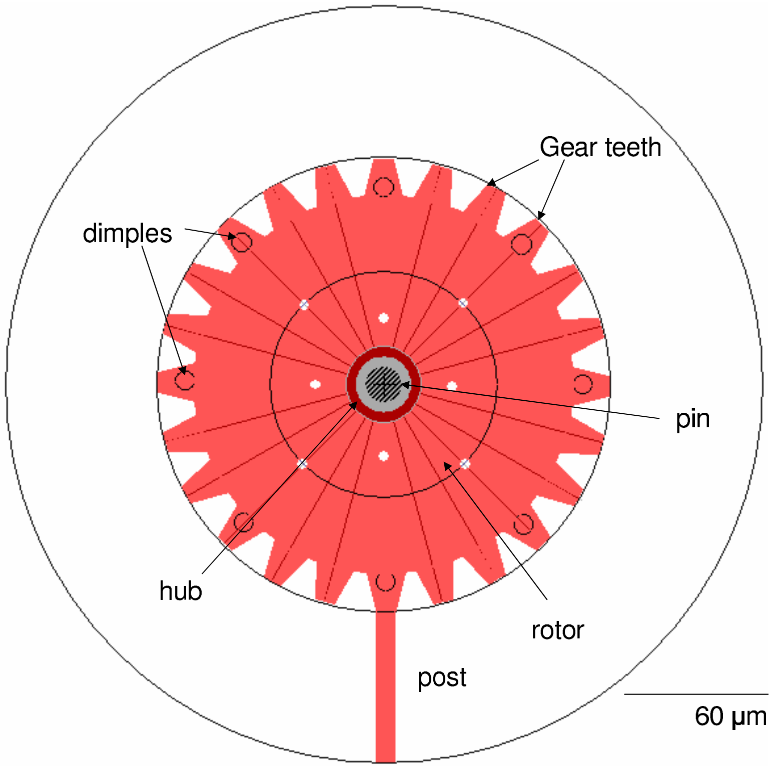


Figure 4.4 Computer aided drawing of the input gear. This is a poly1 gear with dimples on every third spoke that make contact with the nitride layer. The gear rotates about the poly2 pin that is anchored to the substrate. The poly2 hub overlaps the poly1 gear near the pin, causing some poly-poly friction forces.

The moment of inertia, I, of the gear is

$$I = \frac{1}{2} m R_2^2 \ \left[\mathrm{kg} \cdot \mathrm{m}^2\right] \tag{4.22}$$

where m is the mass of the gear and R_2 is the outer radius of the poly1 rotor. The mass is calculated by multiplying the density of polysilicon by the volume of the rotor.

$$m = \rho_{poly} V_{rotor} \ [kg] \tag{4.23}$$

where V_{rotor} is calculated as

$$V_{rotor} = \pi R_2^2 t_{rotor} \ \left[\mathrm{m}^3\right] \tag{4.24}$$

The thickness of the poly1 rotor, t_{rotor}, is 2 μm.

The damping coefficient, c, for the gear is the sum of three different coefficients [40]. These are the coefficient of viscous drag on the top surface of the rotor, c_{d1}, the coefficient of viscous drag for the gap between the pin and the rotor, c_{d2}, and the coefficient of viscous drag from the rotor-nitride gap, c_{d3}. The equations for each of these follows [40]

$$c_{d1} = 0.308 \pi \rho_{air} (R_2^4 - R_1^4)(\theta' \upsilon)^{\frac{1}{2}} \ \left[\frac{\mathrm{kg} \cdot \mathrm{m}^3}{\mathrm{s}^{\frac{1}{2}}}\right] \tag{4.25}$$

$$c_{d2} = \frac{2\pi \eta h_{pin} r^2 R_1}{R_1 - r} \ \left[\frac{\mathrm{N} \cdot \mathrm{m}}{\mathrm{s}}\right] \tag{4.26}$$

$$c_{d3} = \frac{\pi \eta (R_2^4 - R_1^4)}{2h} \ \left[\frac{\mathrm{N} \cdot \mathrm{m}}{\mathrm{s}}\right] \tag{4.27}$$

where ρ_{air} is the density of air in $\frac{\mathrm{kg}}{\mathrm{m}^3}$, R_1 and R_2 are the inner and outer radii of the rotor in m, θ' is the angular velocity of the rotor in $\frac{\mathrm{rad}}{\mathrm{s}}$, υ is the kinematic viscosity of air in $\frac{\mathrm{m}^2}{\mathrm{s}}$, η is the dynamic viscosity of air in $\frac{\mathrm{kg}}{\mathrm{m \cdot s}}$, h_{pin} is the height of the pin in m, and r is the outer radius of the pin in m.

The non-geometric values used for this series of equations are

$\rho_{air} = 1.2\ \frac{\mathrm{kg}}{\mathrm{m}^3}$ [40]

$\upsilon = 1.51 \times 10^{-5}\ \frac{\mathrm{m}^2}{\mathrm{s}}$ [40]

$\eta = 1.8 \times 10^{-5}\ \frac{\mathrm{kg}}{\mathrm{m \cdot s}}$ [40]

By inserting these values into Equations 4.25, 4.26, and 4.27, and summing the individual components, the damping coefficient for the gear becomes

$$\begin{aligned} c &= c_{d1} + c_{d2} + c_{d3} \\ &= (0.004512 \frac{\mathrm{kg}}{\mathrm{m \cdot s^{\frac{1}{2}}}})(R_2^4 - R_1^4)\theta'^{\frac{1}{2}} + (2.2619467 \times 10^{-10} \frac{\mathrm{kg}}{\mathrm{s}})(\frac{r^2 R_1}{R_1 - r}) + \cdots \\ & (37.699 \frac{\mathrm{kg}}{\mathrm{m}^2 \cdot \mathrm{s}})(R_2^4 - R_1^4)\ [\frac{\mathrm{N \cdot m}}{\mathrm{s}}] \end{aligned} \tag{4.28}$$

For the input gear with $R_1 = 7.5\ \mu$m, $R_2 = 60\ \mu$m, and $r = 5\ \mu$m, the resulting damping coefficient is

$$c = (5.85 \times 10^{-20} \frac{\mathrm{kg \cdot m^3}}{\mathrm{s}^{\frac{1}{2}}})\theta'^{\frac{1}{2}} + 4.88 \times 10^{-16} \frac{\mathrm{N \cdot m}}{\mathrm{s}} \tag{4.29}$$

The first term in the damping coefficient of Equation 4.29 introduces a nonlinearity to Equation 4.21, resulting in the following differential equation

$$I\theta'' + A\theta'^{\frac{3}{2}} + B\theta' = T_{input} - T_{friction} \tag{4.30}$$

where A and B represent the constant coefficients of the nonlinear and linear damping terms from Equation 4.29. Equation 4.30 has the form of a nonlinear differential equation that can not be solved analytically.

In order to solve Equation 4.30 numerically, the input and friction torques must be calculated. Just as with the damping coefficient, the friction torque, $T_{friction}$, also has three terms that are added together to give the total resistive torque [40].

$$\begin{aligned} T_{friction} &= \mu_{pn}T_{rotor} + \mu_{pp}T_{rotor} + \mu_{pp}T_{input} \\ &= (\mu_{pn} + \mu_{pp})T_{rotor} + \mu_{pp}T_{input} \;\; [\mathrm{N \cdot m}] \end{aligned} \tag{4.31}$$

where μ_{pn} and μ_{pp} are the coefficients of friction for polysilicon on silicon nitride and polysilicon on polysilicon, T_{rotor} is the resistive torque on the rotor, and T_{input} is the input torque. The torque equations can be written [40]

$$T_{rotor} = \frac{2}{3}\pi\rho_{poly}t_{rotor}g(R_2^3 - R_1^3) \;\; [\mathrm{N \cdot m}] \tag{4.32}$$

$$T_{input} = F_{input}R_{contact} \;\; [\mathrm{N \cdot m}] \tag{4.33}$$

where F_{input} is the input force that is applied to the gear a distance of $R_{contact} = 57.5\ \mu\mathrm{m}$ from the center of the gear. All other terms have been defined previously. Assuming an input force of $F_{input} = 850\ \mu\mathrm{N}$, and inserting Equations 4.32 and 4.33 into Equation 4.31 gives

$$\begin{aligned} T_{friction} = (0.3 + 0.54)(\frac{2}{3}\pi(2.33\frac{\mathrm{g}}{\mathrm{cm}^3})t_{rotor}(9.8\frac{\mathrm{m}}{\mathrm{s}^2}(R_2^3 - R_1^3))) + \cdots \\ 0.54(850\mu N)R_{contact} \;\; [\mathrm{N \cdot m}] \end{aligned} \tag{4.34}$$

Equation 4.30 is solved numerically in MATLAB using the ode45.m function. Resulting plots of angular displacement, velocity, and acceleration with respect to time are plotted in Figure 4.5, with the final time corresponding to an angular displacement of $\frac{\pi}{2}$ radians.

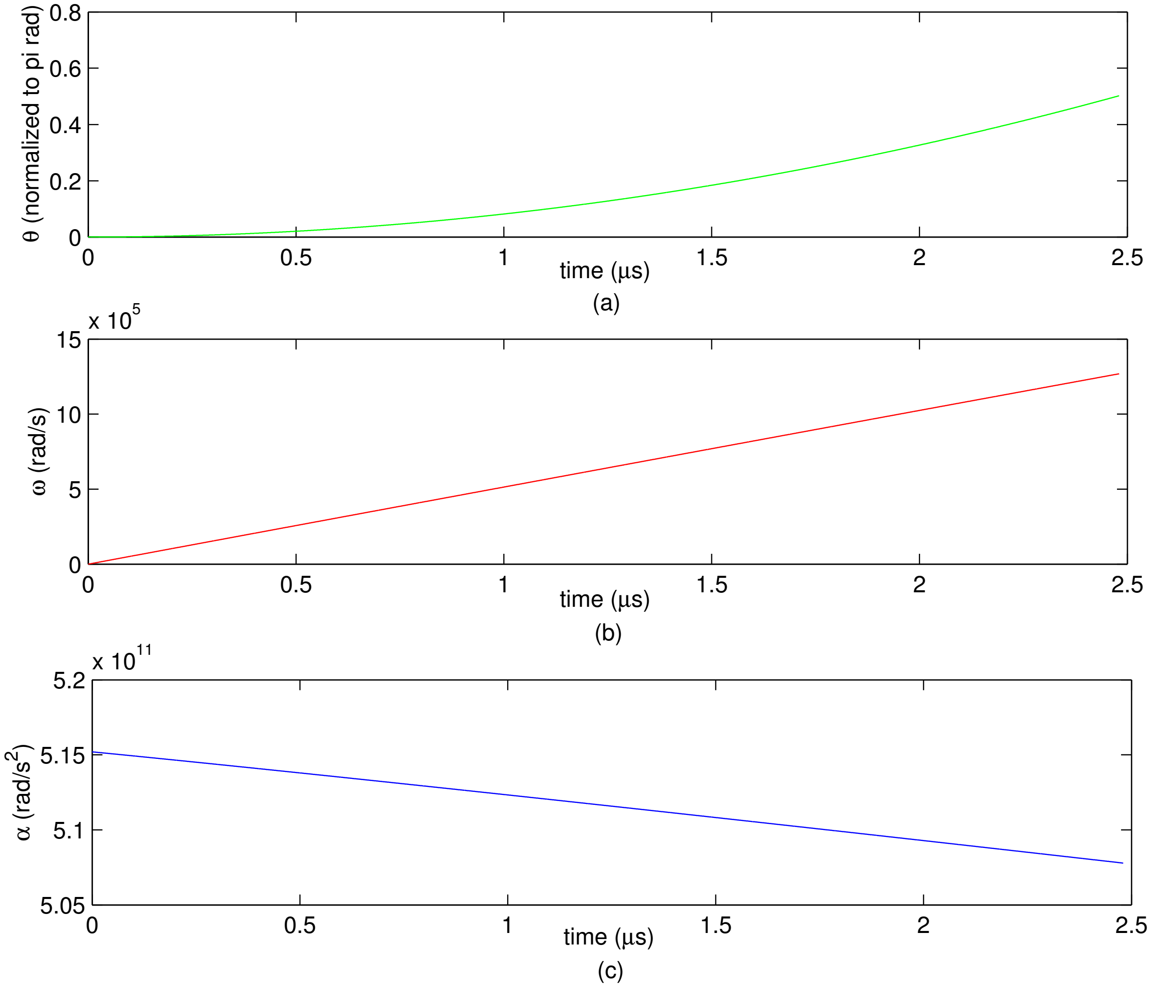


Figure 4.5 Plot of time-dependent behavior of the input gear with a constant input force of 850 μN applied a distance of $R_{contact} = 57.5$ μm from the center of the gear. Values are calculated numerically, and include the nonlinear damping term. (a) Angular displacement vs. time. (b) Angular velocity vs. time. (c) Angular acceleration vs. time.

The significance of the nonlinear term is then examined graphically by plotting the torque contribution that results from including and excluding the nonlinear damping term (Figure 4.6). Based on this analysis, it appears that the nonlinear term can be ignored, since the change in the total torque contribution due to damping is on the order of 10^{-11} N $\cdot$ m, which is three orders of magnitude smaller than the input torque. Neglecting the nonlinear term, and setting $A = 0$ in Equation

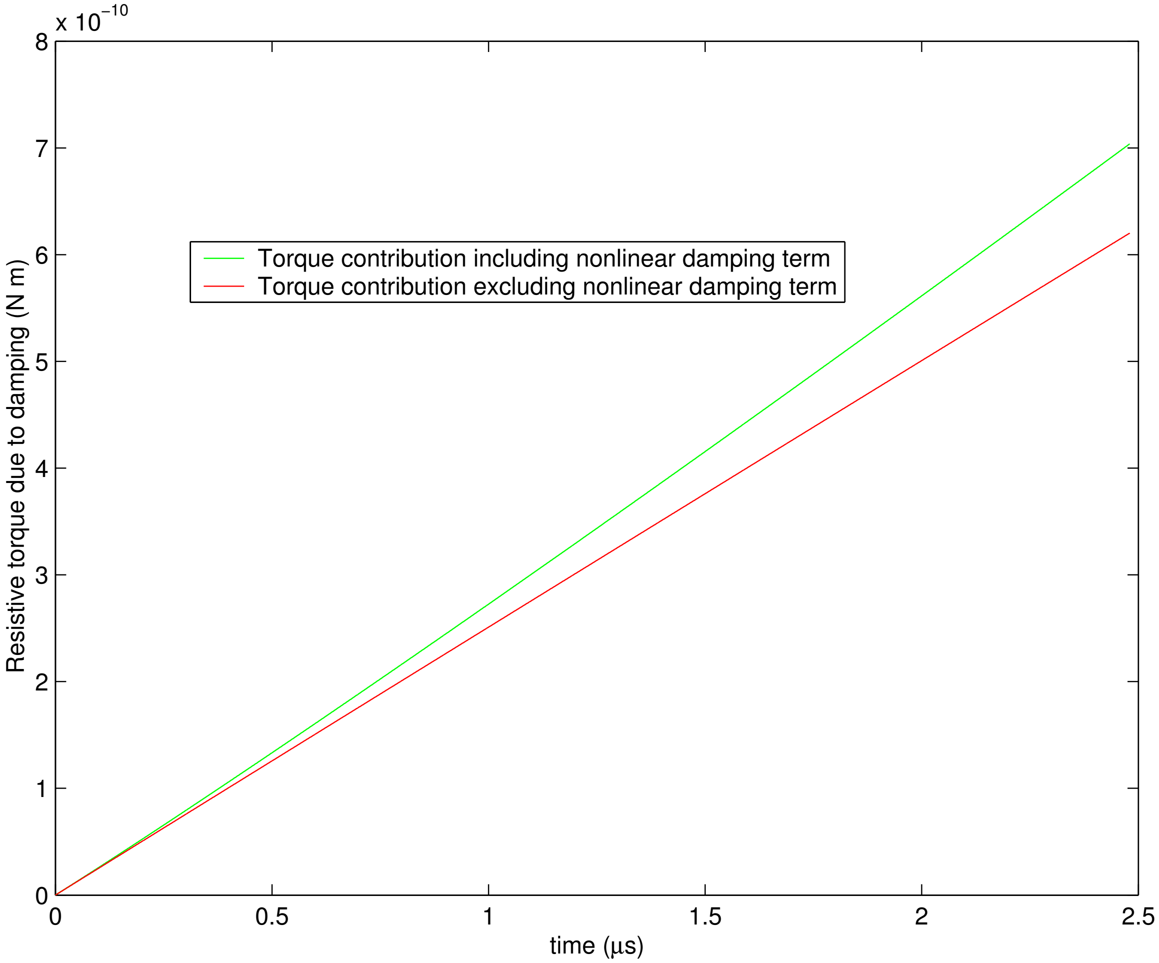


Figure 4.6 Analysis of torque contribution due to damping. The two curves represent the torque contribution due to damping with the nonlinear damping term included and excluded in the calculation.

4.30, results in a linear differential equation, which can be solved analytically.

$$I\theta'' + B\theta' = T_{input} - T_{friction} \tag{4.35}$$

Equation 4.35 can be simplified to standard form as

$$\theta'' + \frac{B}{I}\theta' = \frac{T_{input} - T_{friction}}{I} \tag{4.36}$$

The solution to Equation 4.36 defines angular displacement of the gear with respect to time.

$$\theta(t) = \frac{T_{input} - T_{friction}}{B}\left(\frac{I}{B}e^{-\frac{B}{I}t} + t - \frac{I}{B}\right) \tag{4.37}$$

The time constant associated with the input gear is

$$\begin{aligned} \tau_{InputGear} &= \frac{I}{B} \\ &= 194\mu s \end{aligned} \tag{4.38}$$

Analytical equations for angular velocity and acceleration are calculated by taking the first and second derivatives of Equation 4.37, respectively.

$$\theta'(t) = \frac{T_{input} - T_{friction}}{B}\left(1 - e^{-\frac{B}{I}t}\right) \tag{4.39}$$

$$\theta''(t) = \frac{T_{input} - T_{friction}}{I}\left(e^{-\frac{B}{I}t}\right) \tag{4.40}$$

Figure 4.7 shows a plot of the analytical solution of angular displacement, velocity, and acceleration as a function of time. Comparing Figures 4.7 and 4.5 further justifies the validity of neglecting the nonlinear damping term in modelling the input gear. The input gear has a maximum velocity of 12.7×10^5 $\frac{\text{m}}{\text{s}}$ and rotates a full $\frac{\pi}{2}$ radians in 2.48 μs.

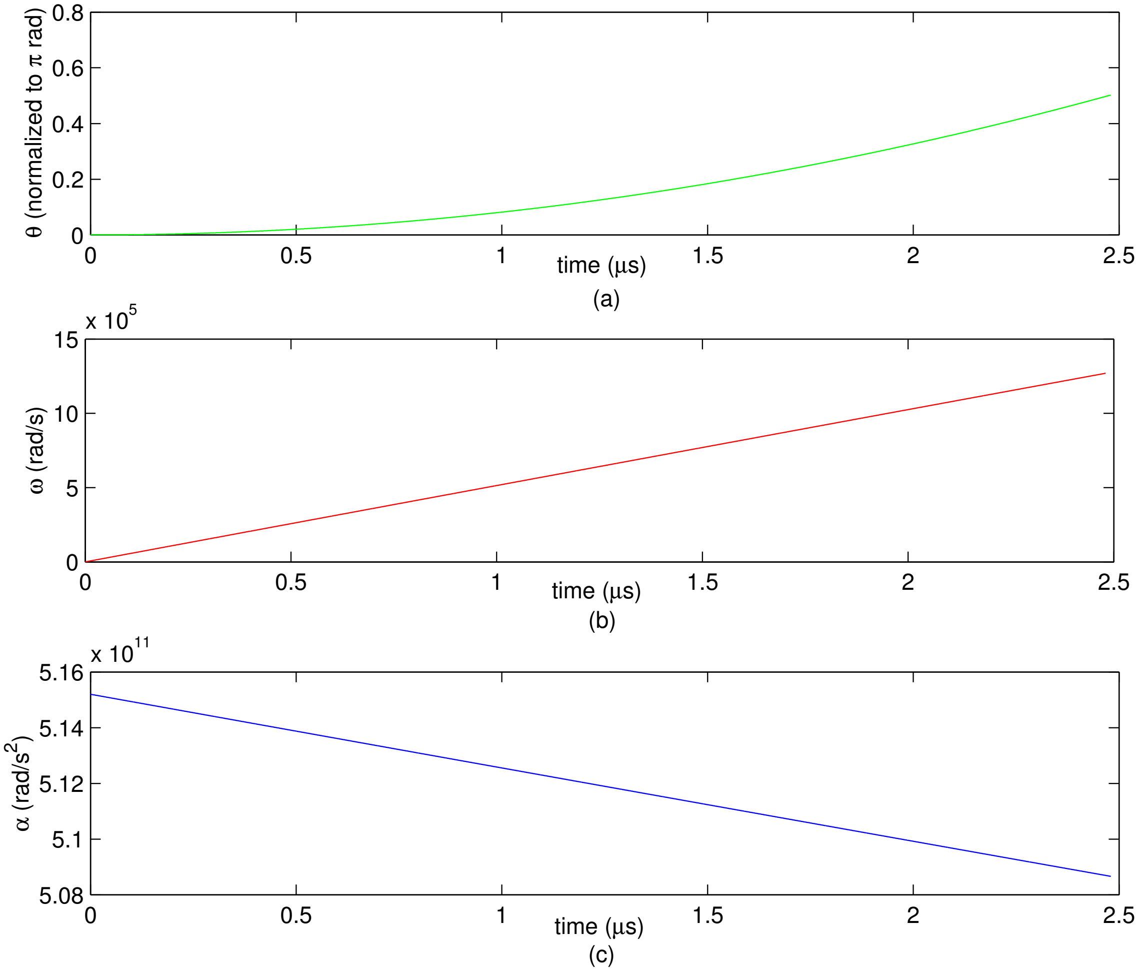


Figure 4.7 Plot of the analytical solution for the time-dependent behavior of the input gear with a constant input force of 850 μN applied at a distance of $R_{contact} =$ 57.5 μm from the center of the gear. Values are calculated analytically by neglecting the nonlinear damping term. (a) Angular displacement vs. time. (b) Angular velocity vs. time. (c) Angular acceleration vs. time.

The transition gear without a restoring spring, as shown in Figure 4.8, is modelled the same as the input gear. With respect to the model, the only difference between the two is their dimensions. The transition gear has $R_1 = 7.5$ μm, $R_2 = 62.5$ μm, $r = 5$ μm, and $R_{contact} = 41.5$ μm. All other parameters are the same. Consequently, it is safe to assume that the nonlinear damping term can be ignored when modelling the transition gear without a spring, as well. Using the analytical solutions for angular displacement, velocity, and acceleration from Equations 4.37, 4.39, and 4.40, the time-dependent behavior of the transition gear without a spring was calculated. Figure 4.9 is a plot of the angular displacement, velocity, and acceleration of the transition gear without a spring up to the maximum displacement of $\frac{\pi}{2}$ radians. The transition gear without a spring has a maximum angular velocity of

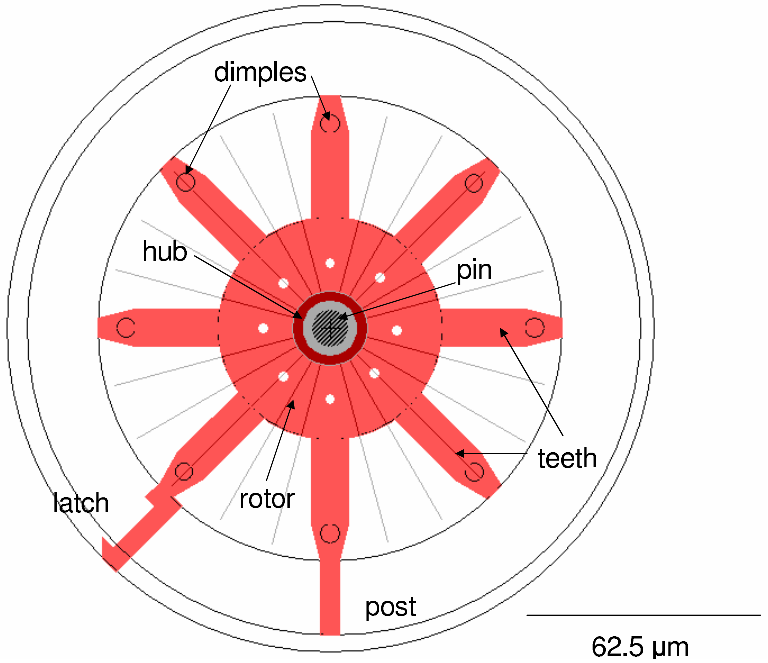


Figure 4.8 Computer aided drawing of the transition gear without a restoring spring. This is a poly1 gear with dimples on all five spokes. The gear rotates about the poly2 pin that is anchored to the substrate. The poly2 hub overlaps the poly1 gear near the pin, causing some poly-poly friction forces.

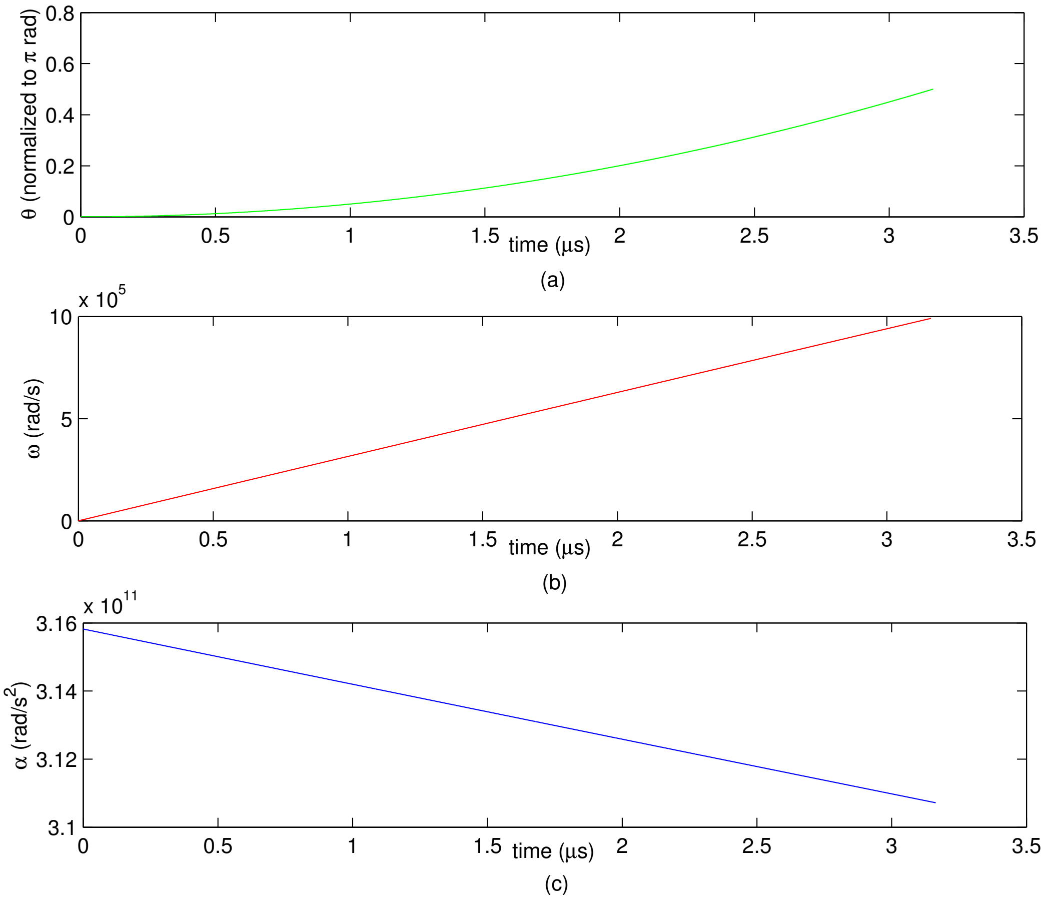


Figure 4.9 Plot of time-dependent behavior of the transition gear without a spring when a constant input force of 850 μN is applied at a distance of $R_{contact} = 41.5$ μm from the center of the gear. An analytical solution was obtained by neglecting the nonlinear damping term. (a) Angular displacement vs. time. (b) Angular velocity vs. time. (c) Angular acceleration vs. time.

$9.9{\times}10^5$ $\frac{\mathrm{rad}}{\mathrm{s}}$ and rotates a full $\frac{\pi}{2}$ radians in 3.16 μs. It has a time constant of

$$\begin{aligned} \tau_{TransitionGearWithoutSpring} &= \frac{I}{B} \\ &= 194\mu s \end{aligned} \tag{4.41}$$

4.1.3.2 Gears with a restoring spring. For the transition gear with a restoring spring (Figure 4.10), Equation 4.21 becomes

$$I\theta'' + c\theta' + G\theta = T_{input} - T_{friction} \tag{4.42}$$

where the rotational spring constant G is included in the equation.

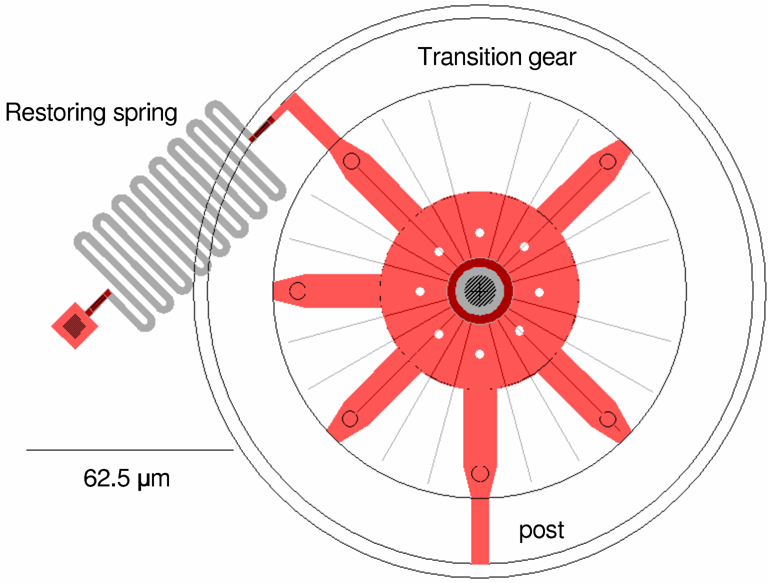


Figure 4.10 Computer aided drawing of the transition gear with a restoring spring. This is a poly1 gear with dimples on all five spokes. The gear rotates about the poly2 pin that is anchored to the substrate. The poly2 hub overlaps the poly1 gear near the pin, causing some poly-poly friction forces. A restoring spring is attached to one of the spokes to cause the gear to return to its original position when released.

The nonlinear damping term is negligible for this gear, as well, and Equation 4.42 can be simplified to a standard second order linear differential equation.

$$\theta'' + \frac{B}{I}\theta' + \frac{G}{I}\theta = \frac{T_{input} - T_{friction}}{I} \tag{4.43}$$

To be consistent with previous derivations, the nonlinear damping coefficient, c, has been replaced with the constant damping coefficient, B. Solving Equation 4.43 results in an analytical solution for angular displacement with respect to time.

$$\theta(t) = \frac{T_{input} - T_{friction}}{I\omega(\alpha^2 + \omega^2)} - \left(\omega e^{-\alpha t}\cos(\omega t) - \alpha e^{-\alpha t}\sin(\omega t) + \omega\right) \tag{4.44}$$

The time constant for the gear with a spring is

$$\begin{aligned} \tau_{GearWithSpring} &= \frac{1}{\alpha} \\ &= \frac{2I}{B} \\ &= 388\mu s \end{aligned} \tag{4.45}$$

Equation 4.44 introduces two new terms, α and ω, where

$$\begin{aligned} \alpha &= \tfrac{B}{2I} \\ \omega &= \tfrac{1}{2I}\sqrt{4IG - B^2} \end{aligned}$$

This was done in order to be able to clearly represent the solution for angular displacement, velocity, and acceleration for the transition gear with a spring. By taking the first and second derivatives of 4.44, analytical solutions for angular velocity and acceleration are calculated.

$$\theta'(t) = \frac{T_{input} - T_{friction}}{I\omega} e^{-\alpha t}\sin(\omega t) \tag{4.46}$$

$$\theta''(t) = \frac{T_{input} - T_{friction}}{I\omega} e^{-\alpha t}\left(\omega\cos(\omega t) - \alpha\sin(\omega t)\right) \tag{4.47}$$

The dimensions for the transition gear with a spring are identical to those of the transition gear without a spring. The only additional value that must be calculated is the rotational spring constant, G. For the transition gear with a spring, the spring that was modelled in Section 4.1.1 with a spring constant of $k_{spring} = 2.74\ \frac{\text{N}}{\text{m}}$ was placed a distance R_{spring} from the center of the gear. The rotational spring constant is

$$G = k_{spring} R_{spring}^2 \ \ [\text{N} \cdot \text{m}] \tag{4.48}$$

After substituting all values into Equations 4.44, 4.46, and 4.47, angular displacement, velocity, and acceleration were plotted with respect to time for a complete rotation of $\frac{\pi}{4}$ radians (Figure 4.11). The maximum angular velocity of the transition gear with a spring is $6.29 \times 10^5\ \frac{\text{rad}}{\text{s}}$, and the gear rotates a full $\frac{\pi}{4}$ radians in 2.3 μs.

4.1.4 Output Lever dynamic response. The final elemental model that must be performed in order to begin creating models for the actual computing devices is that of the output lever. The output lever that was designed for the mechanical logic gates is shown in Figure 4.12.

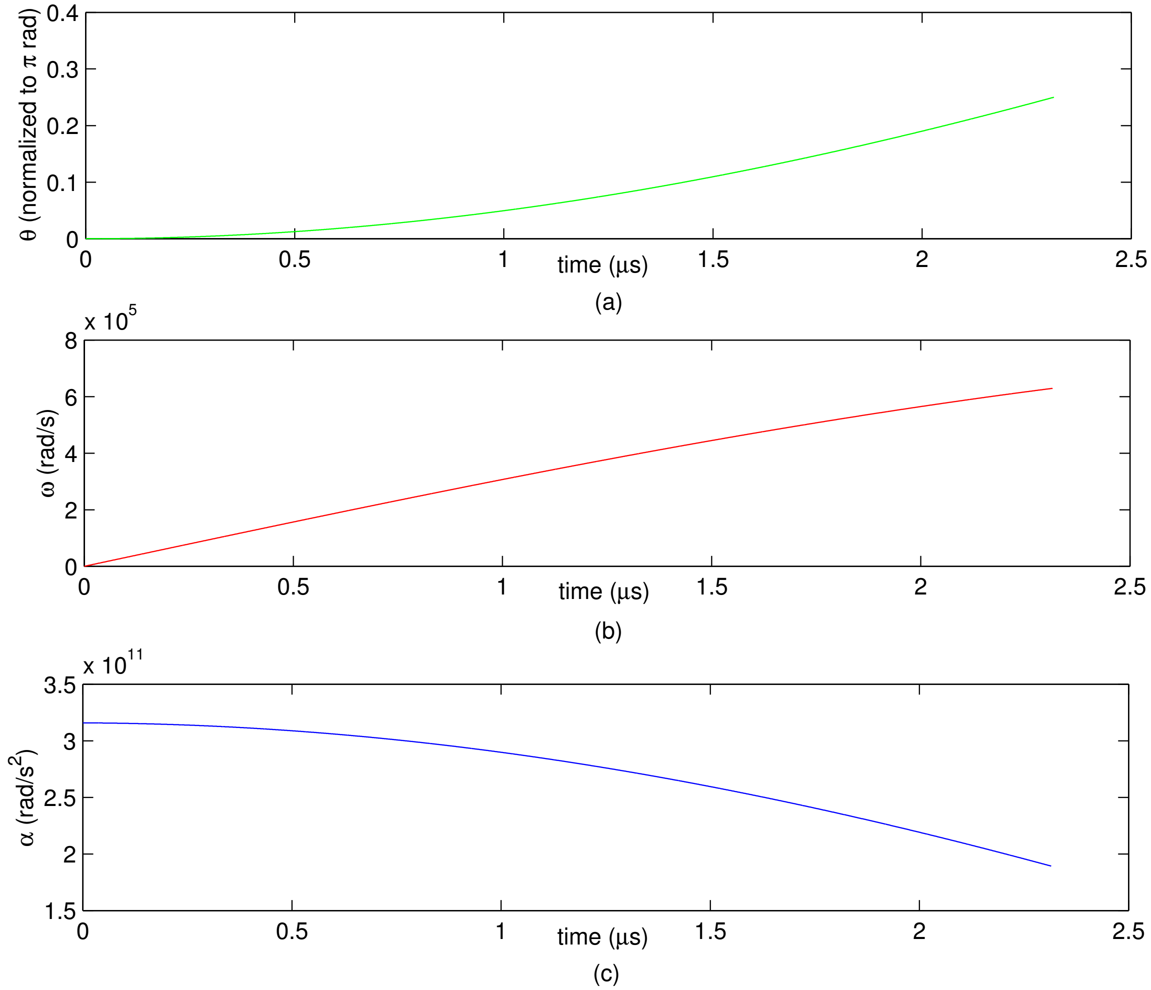


Figure 4.11 Plot of time-dependent behavior of the transition gear with a spring when a constant input force of 850 μN is applied at a distance of $R_{contact} = 41.5$ μm from the center of the gear. An analytical solution was obtained by neglecting the nonlinear damping term. (a) Angular displacement vs. time. (b) Angular velocity vs. time. (c) Angular acceleration vs. time.

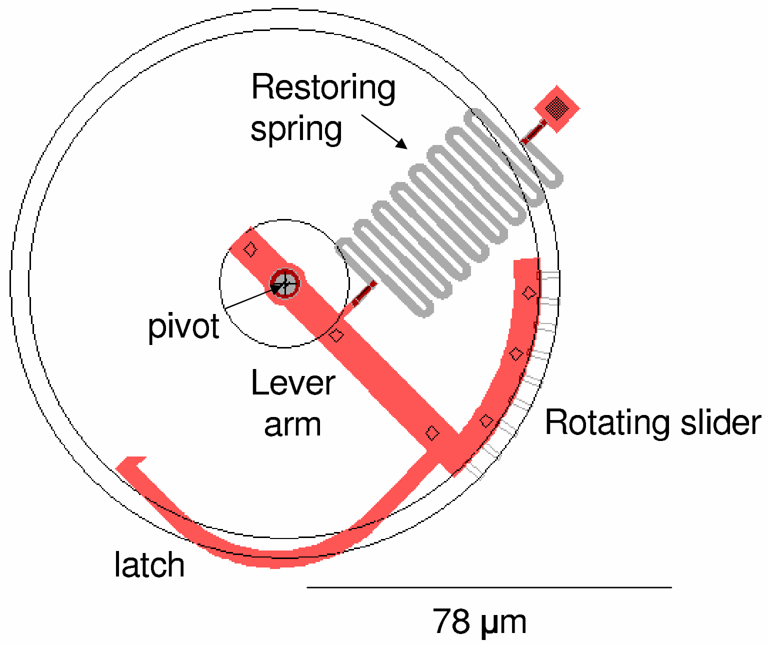


Figure 4.12   Computer aided drawing of the output lever. This is a poly1 output lever with dimples to minimize stiction. The output lever rotates about the poly2 pin that is anchored to the substrate. The poly2 hub overlaps the poly1 pivot near the pin, causing some poly-poly friction forces. A restoring spring is attached to the output lever a distance $R_{spring} = 20 \ \ \mu\text{m}$ away from the pin on the right side of the lever arm. The input force is applied at a distance $R_{contact} = 11 \ \ \mu\text{m}$ away from the pin on the left side. The total length of the right side of the lever arm is $R = 100$ μm.

The output lever arm, which has a spring attached at a distance of R_{spring} = 20 μm from the pivot pin, was modelled as a gear with a spring attached. The dimensions that were used are $R_1 = 3.5$ μm, $R_2 = 78$ μm, $r = 1.5$ μm, and $R_{contact} =$ 11 μm. As with the other elements, the nonlinear damping term can be neglected for the output lever. Consequently, Equations 4.44, 4.46, and 4.47 are used to calculate angular displacement, velocity, and acceleration for the output lever, which were plotted with respect to time for a complete rotation of $\frac{\pi}{2}$ radians (Figure 4.13). The maximum angular velocity of the output lever is 1.335×10^5 $\frac{\text{rad}}{\text{s}}$, and the lever rotates a full $\frac{\pi}{4}$ radians in 8.2 μs. The time constant associated with the output lever is

$$\begin{aligned}
\tau_{OutputLever} &= \frac{1}{\alpha} \\
&= \frac{2I}{B} \\
&= 388\mu s
\end{aligned} \tag{4.49}$$

Using the basic elemental models that have been developed in this section, models of the mechanical computing devices will now be derived. For the device models, the mechanical computing devices will be treated as systems of elements, and the relationship between elements within the framework of each system will be discussed.

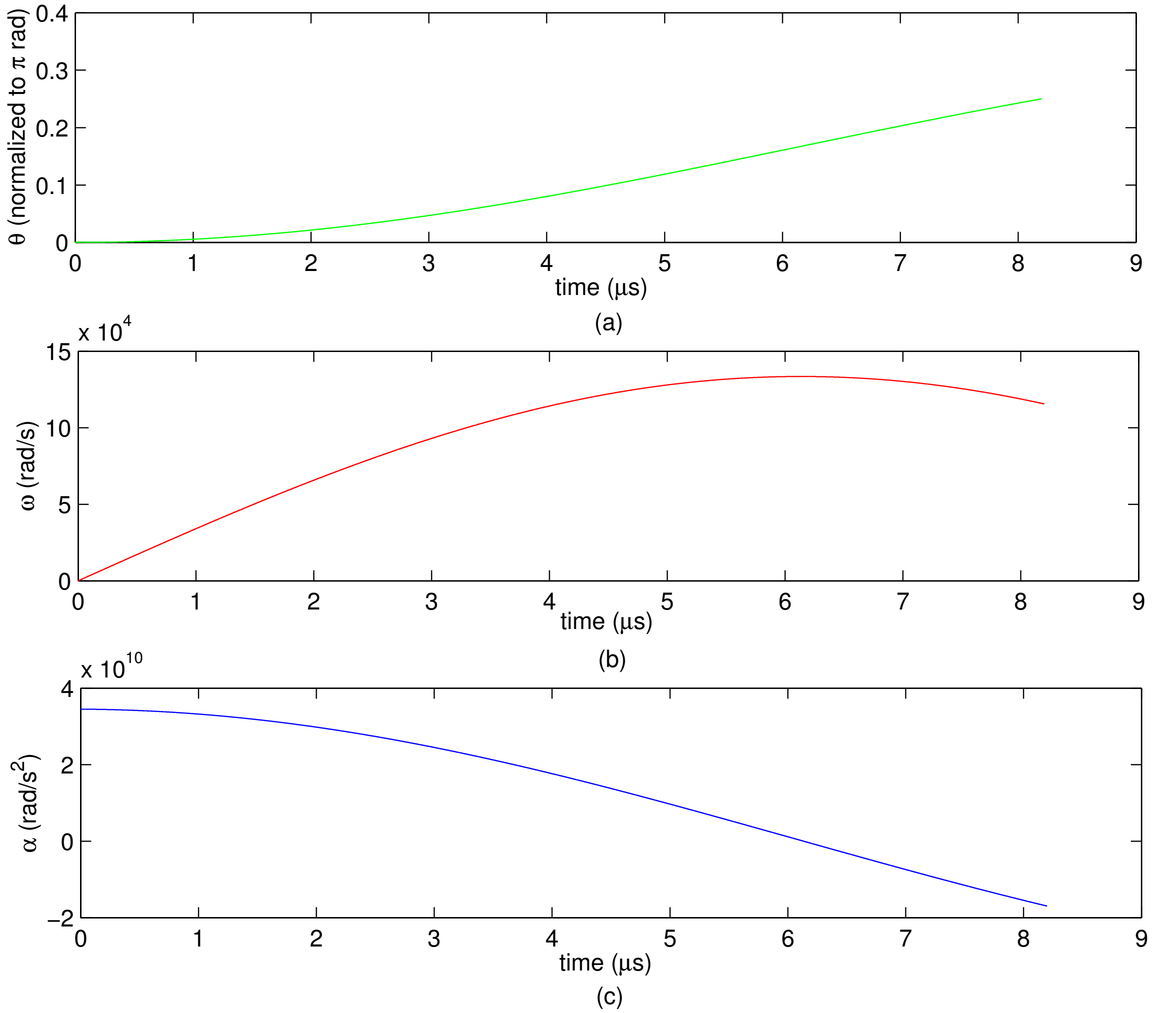


Figure 4.13 Plot of time-dependent behavior of the output lever when a constant input force of 850 μN is applied at a distance $R_{contact} = 11$ μm from the pivot pin. An analytical solution was obtained by neglecting the nonlinear damping term. (a) Angular displacement vs. time. (b) Angular velocity vs. time. (c) Angular acceleration vs. time.

4.2 System Models of Mechanical Computing Devices

Each of the mechanical computing devices that is presented in this section consists of a combination of basic elements. The basic elements were modelled in Section 4.1. The derivations in this section take advantage of the results from the elemental models to produce a realistic model of each device. The main results from the elemental models are presented in Table 4.1 for reference.

Element	displacement	maximum velocity	time
slider	80 μm	96 $\frac{\text{m}}{\text{s}}$	1.66 μs
input gear	$\frac{\pi}{2}$ rad	12.7×10^{5} $\frac{\text{rad}}{\text{s}}$	2.48 μs
transition gear without spring	$\frac{\pi}{2}$ rad	9.9×10^{5} $\frac{\text{rad}}{\text{s}}$	3.16 μs
transition gear with spring	$\frac{\pi}{4}$ rad	6.29×10^{5} $\frac{\text{rad}}{\text{s}}$	2.3 μs
output lever	$\frac{\pi}{4}$ rad	1.335×10^{5} $\frac{\text{rad}}{\text{s}}$	8.2 μs

Table 4.1 Main results from the elemental models.

While the elemental models discussed velocity and switching time for a given element, the system model will also take into account the relationship between interactions of elements to establish true switching times for the devices. The devices that are modelled in this section are inverters, NAND, NOR, and XOR logic gates. The first device to be modelled is the inverter.

4.2.1 Inverter switching speed. As was discussed in Section 3.2.1, an inverter is used to produce an output that is in the opposite state of its input. Figure 4.14 shows the basic elements of the inverter, which are two sliders separated by a gear. For the inverter in Figure 4.14, the input is initially set to 0, and the output is initially set to 1.

The switching speed of the inverter is measured by how fast the output responds to the input. For a constant input force of 850 μN, the switching time for the slider was found to be 1.66 μs, while the switching time for the input gear was found to be 2.48 μs. The maximum of these switching times will be considered the minimum switching time of the inverter, $t_{inverter}$, with the gear acting as the limiting

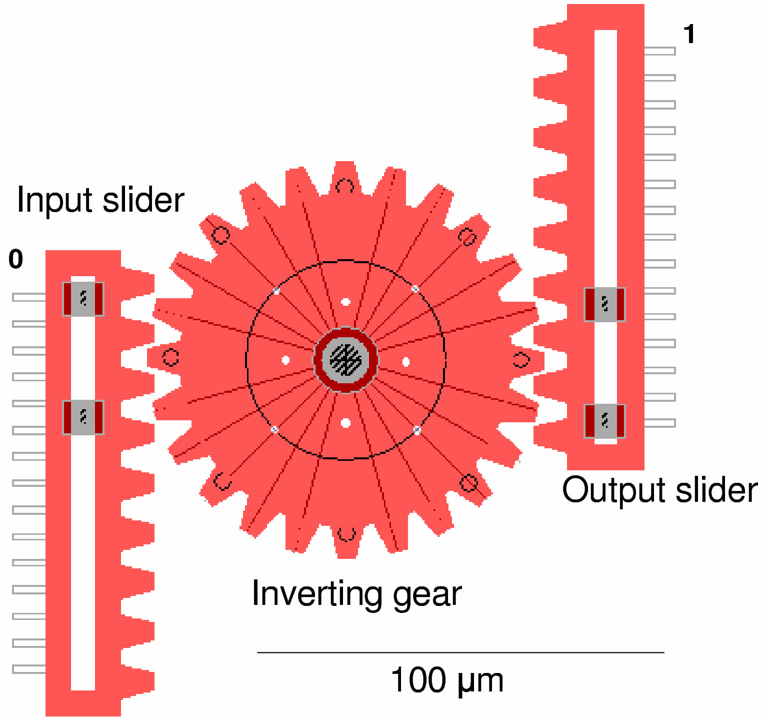


Figure 4.14 Computer aided drawing of the inverter with its input initially set to 0.

element. The minimum switching time for the inverter, $t_{inverter}$ is 2.48 μs, which means that the inverter can switch at a frequency of $f_{inverter} = \frac{1}{2t_{inverter}} = 0.2$ MHz.

4.2.2 NAND gates switching speed. Modelling the logic gates requires a more complete understanding of the device operation. For the logic gates, two switching times will be presented, since switching time for the purely mechanical logic gates is state-dependent. In other words, the switching time depends on the current input state. If both inputs must be switched in order to switch the output, the switching time is longer than if only one of the inputs must be switched.

As discussed in Section 3.2.2, two distinct structures of NAND gates were designed. These are defined as NAND-1 and NAND-0 structures, referring to the initial output state of the device, 1 or 0. The model for each of these structures follows, beginning with the model of the NAND-1 structure.

The NAND-1 gates consist of three sliders, two input gears, a transition gear without a restoring spring, and an output lever. Figure 4.15 shows the NAND-00 logic gate, which will be referred to during the discussion of the model for the NAND-1 gate structure. The switching speed of the NAND-1 gate is measured by how fast the output responds to the inputs. The inputs to the NAND-1 gate cannot move simultaneously. Therefore, assuming both input sliders must be switched to switch the output, as in the NAND-00 case, a switching time can be calculated by adding the time required for each slider movement to propagate to the output. Switching the first input slider to the 1 position is limited by the combination of the input gear and the transition gear without a spring. The input gear must rotate $\frac{\pi}{2}$ radians, and push the transition gear without a spring $\frac{\pi}{4}$ radians. The transition gear will begin moving from rest after the input gear has already rotated $\frac{\pi}{4}$ radians. It takes the input gear 1.75 μs to rotate $\frac{\pi}{4}$ radians. The transition gear must then rotate

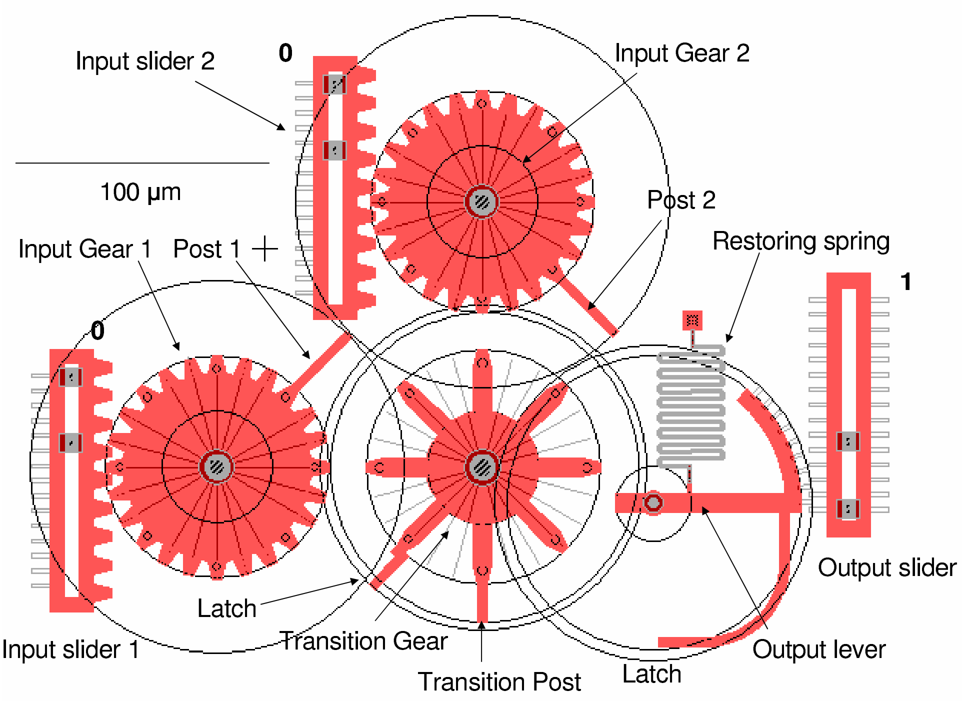


Figure 4.15   Computer aided drawing of the NAND-1 gate structure. Both inputs of the gate structure are initially set to 0 (NAND-00 gate).

$\frac{\pi}{4}$ radians. It does so, beginning from rest, in 2.235 μs. The total time required to switch the first input is 3.985 μs. To minimize the switching time, the second input is switched immediately following the first. This occurs in an additional 3.985 μs. However, the output lever requires an additional 8.2 μs to switch the output state. Therefore, the total switching time required to switch both inputs and the output is 16.17 μs. Switching from a state that requires only one of the inputs to be switched takes 12.185 μs. In order to represent the switching speed for this device in frequency, an average switching time of 14.18 μs will be used. This corresponds to a frequency of 35.26 kHz.

The switching speed of the NAND-0 gate is faster than that of the NAND-1 gate, since both inputs can be switched simultaneously. The NAND-0 gate structure consists of three sliders, two input gears, a transition gear with a restoring spring, and an output lever. Figure 4.16 shows the NAND-11 logic gate, which will be referred to during the discussion of the model for the NAND-0 gate structure. For the NAND-0 structure, switching from the 110 state to the 001 state (assuming both inputs are switched simultaneously), requires a time of 1.75 μs for the input gears to rotate $\frac{\pi}{4}$ radians. The transition gear without a spring must then rotate $\frac{\pi}{4}$ radians, which it does in 2.315 μs, while the input gears complete their total rotation of $\frac{\pi}{2}$ radians. Finally, the output lever must rotate $\frac{\pi}{4}$ radians, which takes 8.2 μs. The total time required to switch states is 12.265 μs. This is true for switching that involves switching one or both inputs. The maximum frequency for switching the NAND-0 gate is 40.8 kHz.

The NAND gate switching speed is significantly limited by the output lever. This is due to the fact that the output side of the output lever is five times as long as the input side. This allows for the output slider to have comparable movement to the input sliders, but results in a slower device with an output force one-fifth as large as the input force. Systems of logic gates that are combined in series experience significant fan-out due to this force loss. This idea will be revisited in Section 4.2.4

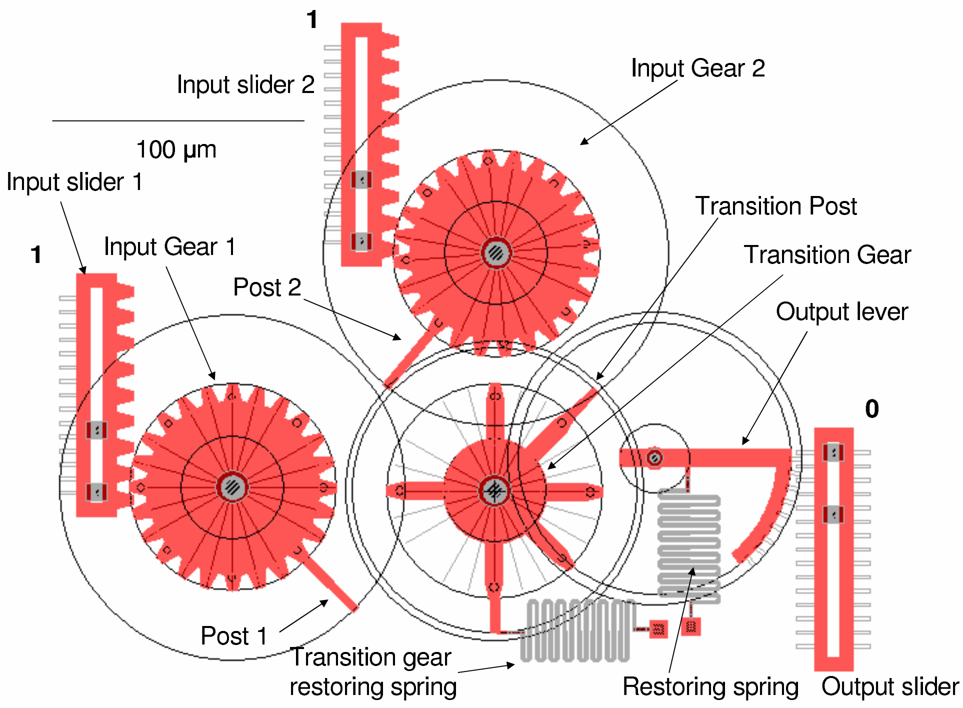


Figure 4.16 Computer aided drawing of the NAND-0 gate structure. Both inputs of the logic gate are initially set to 1 (NAND-11).

when XOR gates are modelled as combinations of inverters and NAND gates. Before discussing the XOR gates, an analysis of the NOR gates is presented.

4.2.3 NOR gates switching speed. As discussed in Section 3.2.3, the NOR-0 and NOR-1 structures are identical to the NAND-1 and NAND-0 structures, respectively. The NOR gate is a physically inverted version of a NAND gate. Consequently, the modelling results that were calculated for the NAND gates apply to the NOR gates, and will be repeated here for clarification. The NOR-0 gates consist of three sliders, two input gears, a transition gear without a restoring spring, and an output lever. Figure 4.17 shows the NOR-11 logic gate. Since the inputs can not be applied simultaneously, this device has two switching times. The first is for switching from the 110 state to the 001 state. This requires a time of 16.17 μs. Switching between any other states takes 12.185 μs. This results in an average switching time of 14.18 μs, which corresponds to a frequency of 35.26 kHz.

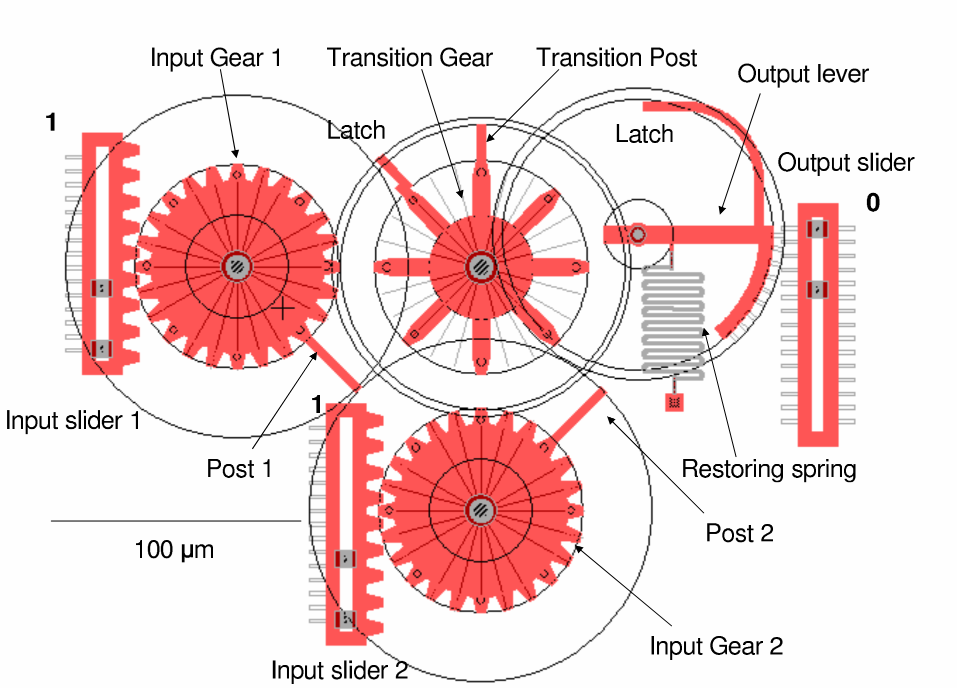


Figure 4.17 Computer aided drawing of the NOR-0 gate structure. Both inputs of the gate structure are initially set to 1 (NOR-11 gate).

The NOR-1 gate, which has the same structure as the NAND-0 gate, consists of three sliders, two input gears, a transition gear with a restoring spring, and an output lever. Figure 4.18 shows the NOR-00 logic gate. As with the NAND-11

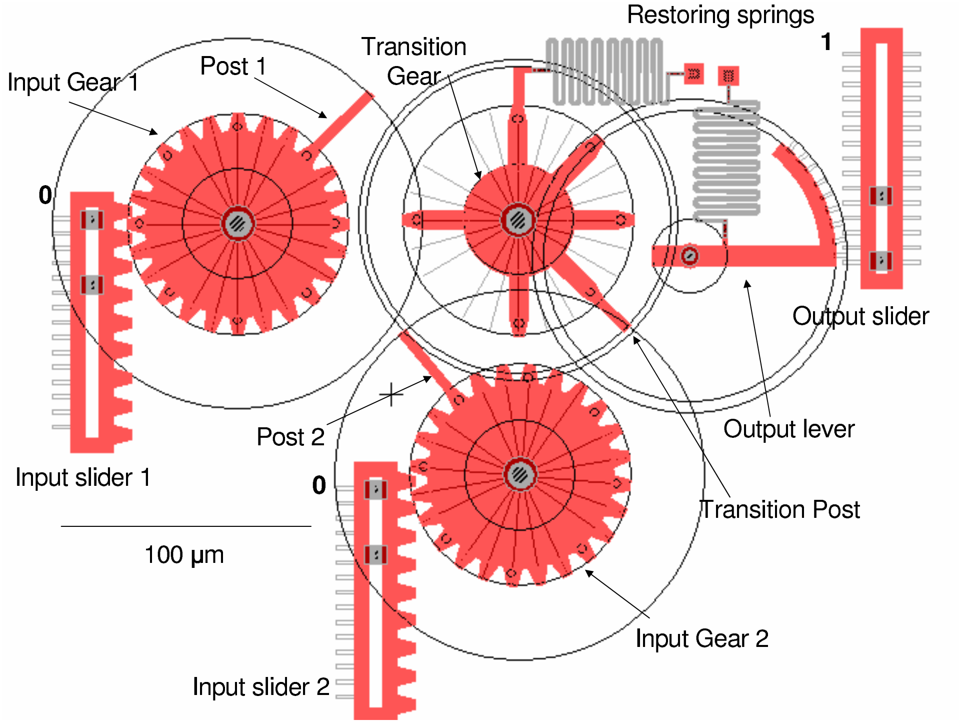


Figure 4.18 Computer aided drawing of the NOR-1 gate structure. Both inputs of the gate structure are initially set to 0 (NOR-00 gate).

logic gate, since the inputs can be applied simultaneously, the NOR-1 gate has only one switching speed, with a switching time of 12.265 μs, which corresponds to a maximum frequency of 40.8 kHz.

4.2.4 XOR gates switching speed and input force fanout. The XOR gates use a combination of two inverters and three NAND gates, with the output of two NAND gates serving as the input to the third. All four XOR gate configurations use two NAND-1 structures and one NAND-0 structure. For the case of the XOR-0 gates, two NAND-1 gates are used as inputs to a NAND-0 gate. For the XOR-1 gates, a NAND-1 gate is used as output to a NAND-0 and NAND-1 gate input combination. The switching speed is limited to the switching speed of the NAND gates. Since the inputs can not be applied simultaneously, each of the input NAND gates has a switching time of 14.18 μs, and, assuming no loss is experienced between the input and output, with exception of the drop of output force to one-fifth that of the input force due to the output lever, a force of 170 μN is applied to the output NAND gate. Again, assuming that the only force loss is due to the output lever of the output NAND gate, an input force of 170 μN is applied to the output lever. This results in a maximum angular displacement of 0.067π radians. An angular displacement of $\frac{\pi}{4}$ is required for correct device operation. For this condition, a minimum input force of 635 μN must be applied to the output lever. Again, assuming no loss in force, other than the losses due to the output levers, an input force of 3.175 mN must be applied to the input sliders of the XOR gate for the output slider of the XOR gate to move appropriately. Switching time for the XOR gate with an input force of 3.175 mN applied to each of the input sliders is 37.31 μs, which corresponds to a switching frequency of 13.4 kHz. The output force of the XOR gate is $\frac{F_{input}}{25}$, or 127 μN, which is probably not enough force to serve as an input to another device. Therefore, fan-out is a critical issue with respect to purely mechanical computing.

V. Experimental Procedures

This chapter discusses the experimental procedures that were followed, beginning with the layout of the designs to be fabricated and terminating with the collection of data in the form of pictures, video, and scanning electron microscope (SEM) images. The discussion of device testing refers back to Chapter III since the operation of each devices has already been discussed in detail. Consequently, testing is performed to determine if the devices operate as designed. After the designs have been submitted to be fabricated, post-processing of the fabricated devices is required before operational testing can begin. The required post-processing procedures are discussed first.

5.1 Post-Processing

The designs for this thesis were layed out in 1900 μm x 1900 μm sample regions, 25 of which were placed in a 1 cm x 1 cm grid. The 1 cm x 1 cm design layout was then fabricated using the PolyMUMPs process and covered with a protective photoresist layer. The protected 1 cm x 1 cm chip was then diced into 25 2 mm x 2 mm die, which are ready to be released and tested. The release process is discussed as required post-processing, along with any device-dependent steps that must be taken prior to device operation. Testing is discussed in Section 5.2.

5.1.1 Release Process. Six die are taken at once into the AFIT Class 10,000 Clean Room to release the MEMS devices. Due to the toxic nature of certain chemicals that are used as part of the release process, the entire process, with the exception of drying the die, is performed at the acid bench. Protective equipment, including a lab apron, gloves, goggles, and a protective masking shield are worn during the process.

After cleaning three glass petri dishes, two 50 ml plastic beakers, and a metal die basket with acetone, methanol, and deionized (DI) water, the containers are placed under the hood at the acid bench. Labels specifying contents to be placed in each container are written next to each container. The first petri dish is filled with 25 ml of acetone. The second is filled with 25 ml of methanol. Next, the first plastic beaker is filled with 50 ml of 48% hydroflouric acid (HF), and the second with a 3:1 mixture of methanol and DI water. 15 ml of methanol and 5 ml of DI water is sufficient. The final petri dish is filled with 25 ml of methanol and the bottom of the die holder is set inside to soak. Figure 5.1 shows a picture of the experimental setup for the release process. All six die are immersed in the acetone for 15 minutes followed by 5 minutes in methanol to remove the protective photoresist layer from the samples. Each die is then immersed, one at a time, in a solution of 48% HF for 4 minutes to etch the sacrificial oxide layers and release the structural polysilicon layers. After each die has been exposed to the HF for 4 minutes, it is dipped in the

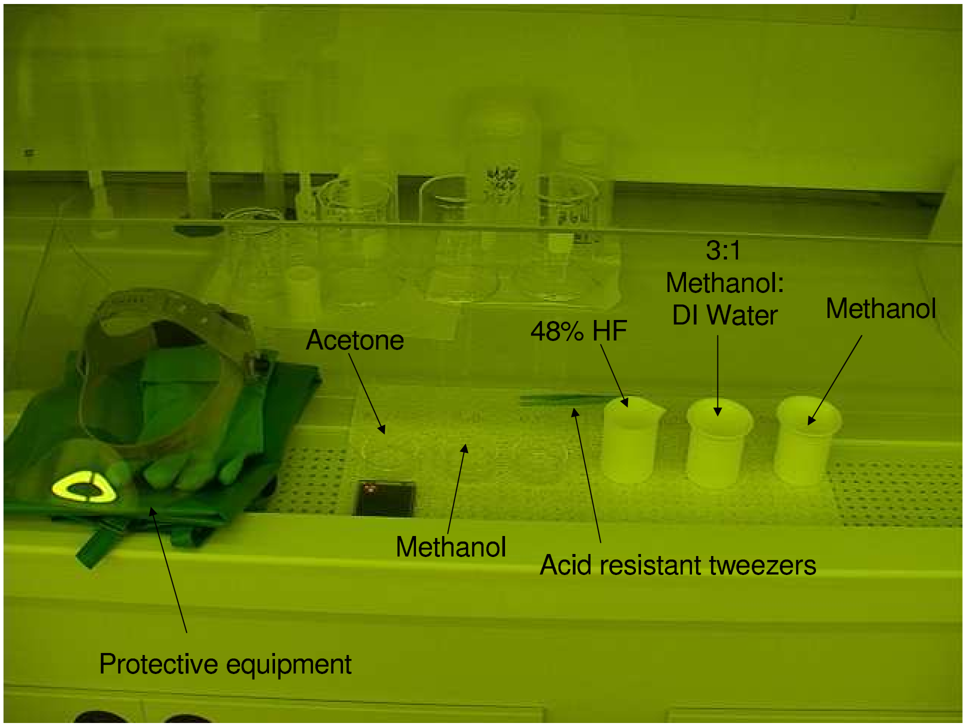


Figure 5.1 Experimental Setup for the release process in the clean room.

solution of 3:1 Methanol:Water, and then placed in the die holder, which is soaking in enough methanol to completely immerse the die until all die have completed the process. The methanol solution containing the six released die are then taken to the Autosamdri® -815 automatic critical point CO_2 dryer shown in Figure 5.2. The Autosamdri® -815 is filled with enough methanol to cover the samples when the metal die container is transferred from the petri dish into the chamber. The Autosamdri® -815 is used to dry the samples to prevent or lessen the likelihood of stiction [45]. With the exception of the purge cycle, which is manually set to a 5 minute cycle by placing the Purge Time Control to position 1, the entire drying process is performed automatically by the Autosamdri® -815. The entire drying cycle takes approximately 30 minutes. At the conclusion of the drying cycle, the metal die container is removed from the Autosamdri® -815, and the six die are taken out of the container and placed in a gel-pack container to be transported to the test lab for operational testing and data capture. In the test lab, further post-processing is

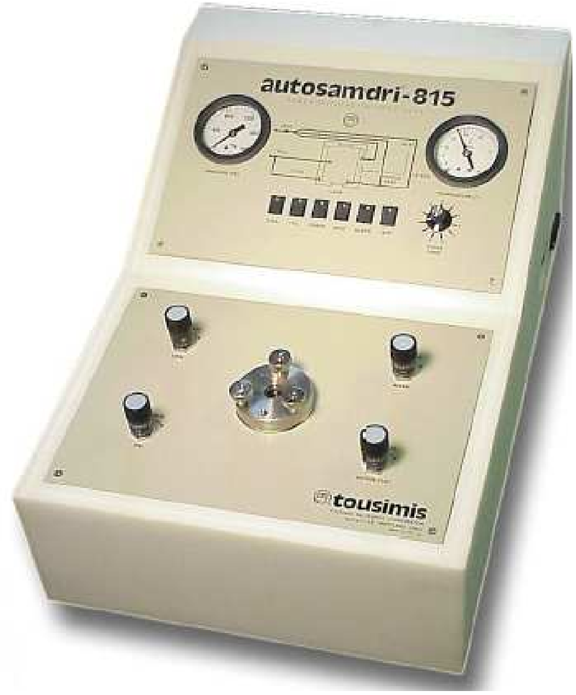


Figure 5.2 Autosamdri® -815 critical point CO_2 dryer.

performed (if necessary) prior to testing and capturing video images and SEMs of the devices.

5.1.2 Device specific post-processing. As discussed in Chapter III, some devices require additional post-processing prior to devices being classified as operational. This is due to the limitations of the PolyMUMPs process with respect to the conformal nature of the polysilicon structures, and the desire to have planarized layers. Among these devices are the integrator and the trigonometric functions. The required post-processing for each of these devices, as discussed in Chapter III entails sliding one part of the computing device into the correct position with respect to the other part. This is performed under a microscope at the Micromanipulator Probe Station, shown in Figures 5.3 and 5.4. Once all post-processing is completed, the

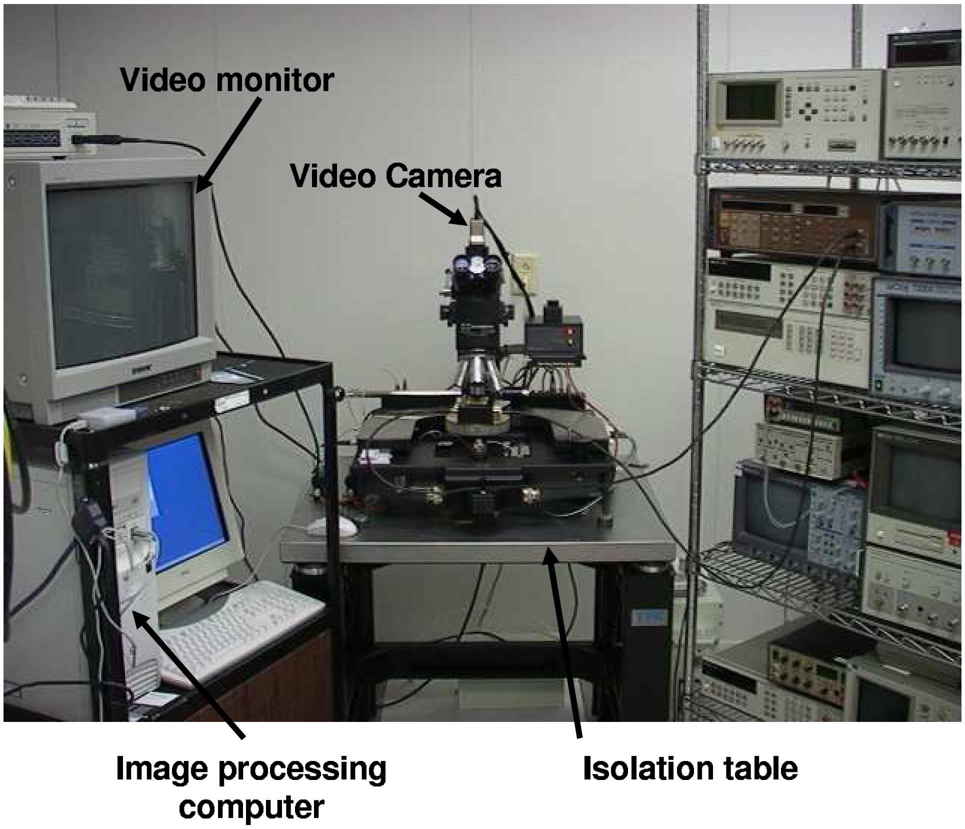


Figure 5.3 Probe Station with camera and computer for video and image capture and processing.

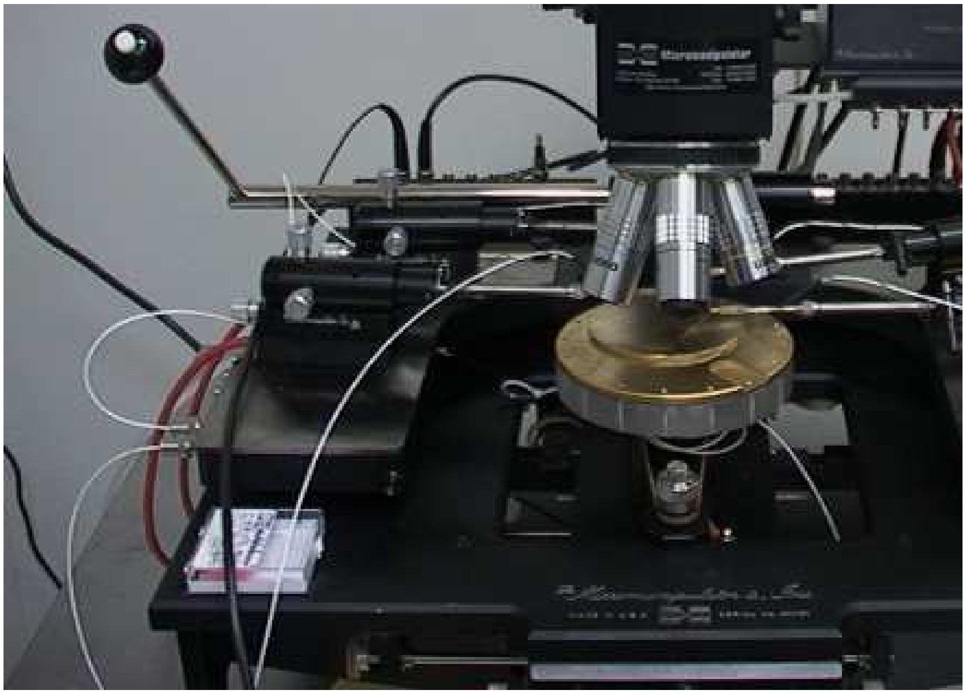


Figure 5.4 Close up of probe station. Microscope and probes are clearly visible.

devices are ready to be tested.

5.2 Testing and data capture

Testing of the purely mechanical computing devices consists of physically moving inputs as discussed in Chapter III and reading the outputs to determine if the device operates as designed. While testing is performed, the video capture system is set to record data in the form of video and images. For devices that do not operate as intended, the scanning electron micrographs (SEMs) can be used to analyze, in great detail, small sections of a device that may not have been designed or fabricated properly. This section discusses the testing process, including video and SEM image capture.

5.2.1 Testing.

The first step in data capture involves testing of each of the devices. All devices have been designed to perform specific functions. During the process of testing, the video capture system is set up to record inputs and outputs, enabling subsequent analysis of data. While recording video, the inputs, whether they are sliders or angle selectors, are moved to a specific input value. The output to the function is then read using rulers that were fabricated alongside the devices. Assuming the correct output is seen, the input value is adjusted for further testing. If a given input results in an unexpected output, further analysis is done to determine if there has been a design flaw or fabrication error. While the video capture equipment helps in capturing input and output relationships for the devices, SEMs are often necessary to see the small details of a device. This is especially useful for locating and documenting problems with a device that is not operating as expected. The video capturing and scanning electron microscope imaging are discussed next, as these form an essential part in testing and analyzing the computing devices.

5.2.2 Video Capture. The video capture system consists of an Emcal Scientific, $\frac{1}{3}$ inch color video camera that is mounted on the probe station, a video acquisition card, and the Ulead® VideoStudio 5 software package for video and image processing. Once a device is placed under the microscope of the probe station, and in position for testing, video is recorded of the device in a ".vsp" (VideoStudio Project) file. Testing is performed, as discussed previously, and the video is stopped. On playback, time markings are recorded for sections of the video that are significant for analysis, and these are either saved as ".mpeg" video files or ".bmp" image files, depending on whether a video of operation or an image of the device is desired. While the video recording helps in analysis, the potential magnification of the image is less than that obtained using the scanning electron microscope. For examining very small details of a device, SEMs are used. Scanning electron microscope imaging will now be discussed.

5.2.3 Scanning Electron Microscope Imaging. For this research, the International Scientific Instruments WB-6 Scanning Electron Microscope (Figure 5.5) is used to obtain SEMs of predominately small sections of devices. These images

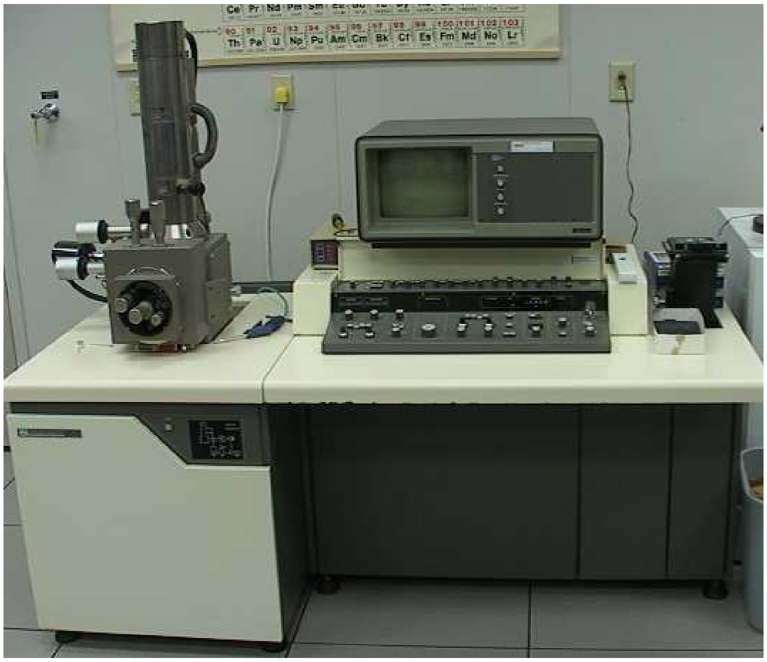


Figure 5.5 International Scientific Instruments WB-6 Scanning Electron Microscope (SEM).

are captured on Polaroid Polapan 55PN black and white film, and treated with a protective coating for preservation. Large device images tend to be of better quality using the video capture equipment, and are preferred in situations where an image of the entire device is desired.

With a combination of video, video images, and SEMs, it is possible to thoroughly analyze the devices that were designed and fabricated. Chapter VI presents the results and analysis of the devices.

VI. Results

This chapter presents the results of the devices that were tested for correct operation. As mentioned previously, some devices that were designed for this research have not been fabricated yet. The results of the devices that have been fabricated and tested are presented here, along with an analysis of the design errors associated with devices that do not work.

6.1 Trigonometric functions

Sine, cosine, and tangent/cotangent function devices were fabricated, released, and tested for correct operation. The sine and cosine function were both successful designs, and operated as expected. Figure 6.1 is a video image of the cosine function prior to being tested. The assembly post-processing step was more challenging than expected, but once the pieces were properly assembled, the sine function and cosine function performed well for angles between 15 and 60 degrees from the vertical and horizontal, respectively. For the cosine function device, the poly2 hub on the angle selector prevented the poly2 buckle from moving any further left than the hub.

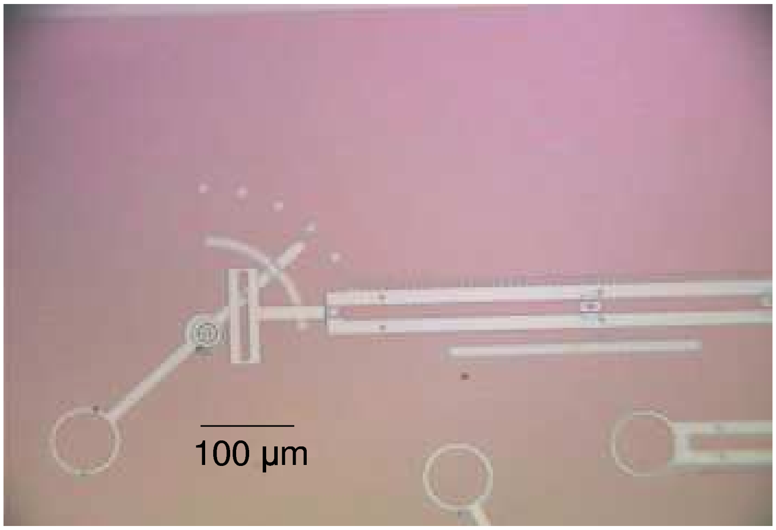


Figure 6.1 Video image of cosine function prior to testing.

Consequently, the angle selector was prevented from being positioned at an angle greater than 60 degrees. Just before reaching an angle of 15 degrees, the buckle broke off the slider, due to torque that results from the angle selector as it is moved between the buckle and the poly0 angle selector guide. Similar results were found for the sine function, which is fabricated as a nearly identical structure. Table 6.1 presents sample data values and the corresponding expected values for the cosine function. The measured values were obtained by opening captured video images in PaintShopPro and zooming in on the image until pixels are clearly visible. The size of one pixel for a given microscope setting was determined in advance of taking measurements, and this conversion factor was used to convert from pixel length to metric.

Angle (degrees)	Experimental Data Value (μm)	Calculated Value
18	53.1	53.7
30	48.75	49
45	40	40

Table 6.1 Experimental data and calculated values for the cosine function.

The tangent/cotangent device (Figure 6.2) did not operate as expected, but the design error was found and can be very easily corrected for a future fabrication run. The slot in the angle selector for the tangent/cotangent function is formed by two very narrow (2 μm) poly1 bars, as shown in the SEM of Figure 6.3. Due to the highly elastic nature of the thin bars, it is not only difficult to place the pin from the slider into the slot, but once the pin is in the slot, any amount of force, such as that caused by moving the angle selector, causes the beam that is being contacted by the pin to bend slightly, allowing the pin to fall out of the slot. Thicker beams are included in the new tangent/cotangent design, so that the bars will not be as elastic, and the device will operate properly.

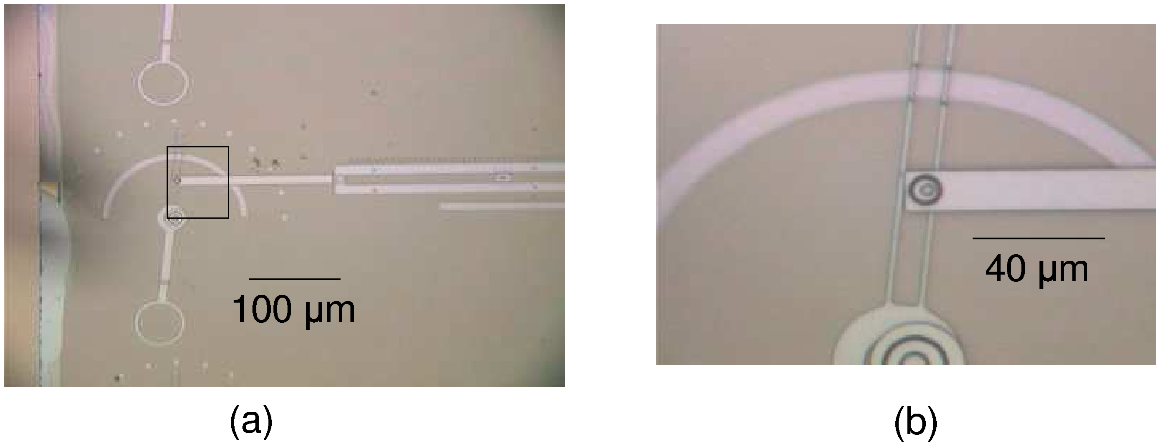


Figure 6.2 Video image of the tangent/cotangent function device after assembly post-processing. (a) Device view. (b) Close-up of knob.

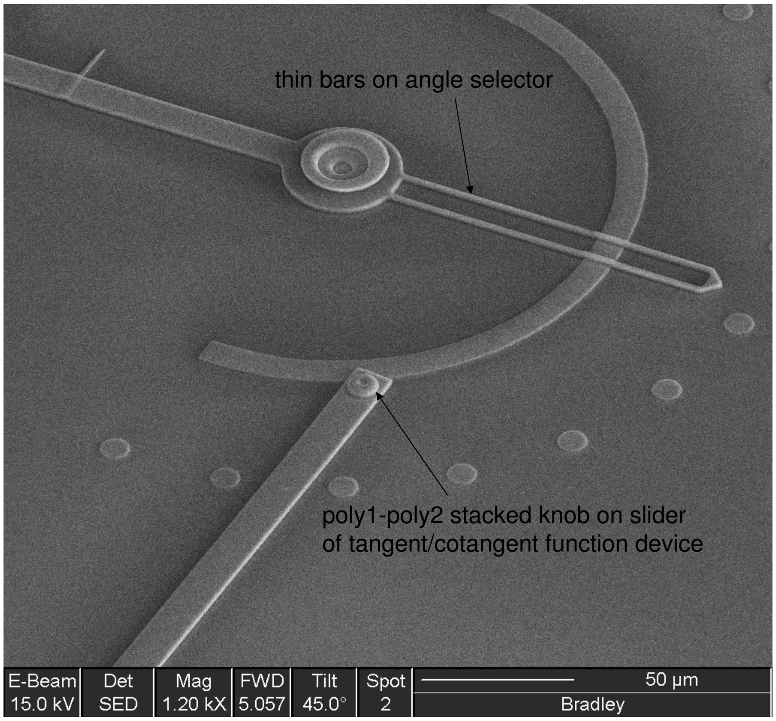

thin bars on angle selector
poly1-poly2 stacked knob on slider
of tangent/cotangent function device
E-Beam 15.0 kV
Det SED
Mag 1.20 kX
FWD 5.057
Tilt 45.0°
Spot 2
50 µm
Bradley

Figure 6.3 SEM of tangent/cotangent device prior to the assembly postprocessing step. The thin bars on the angle selector and the poly1-poly2 stacked knob on the slider is shown. The poly2 slider with the poly1-poly2 stacked knob was designed to be positioned from the top, but inserting the knob from the underside of the angle selector is another viable option.

6.2 Differential

Although a new differential function has been designed (see Figure 3.14) that should function as desired, it will not arrive prior to defending this research. An inferior design is analyzed here, instead. The problem with the inferior differential is that the entire structure consists of one rigid body. At the time the design was conceived, it was assumed, in error, that one input slider could be moved, causing the linkage to flex and move the output slider without affecting the other input. This was not the case, however. In fact, the entire body was so rigid that input movement was not possible without moving the entire structure. Figure 6.4 shows an SEM of the failed differential.

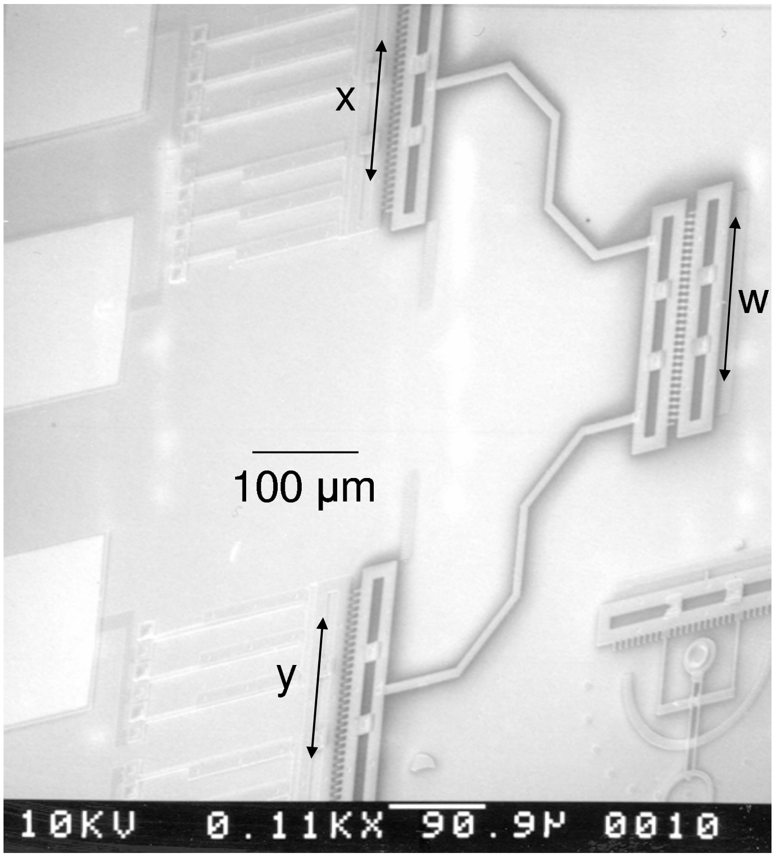


Figure 6.4 SEM of a differential. The output and inputs form a rigid body. The device does not work.

6.3 Multipliers

The multiplier does not work as designed. However, the design problem was found, and an improved device will be fabricated along with the other devices, although it will not be tested prior to the defense of this thesis. The design error involved the incorrect connection of the x-input with the output. Rather than being connected by the conformal overlap of the poly2 x-input on the poly1 output (as was done for the connection of the y-input to the diagonal crossbar), both were fabricated as poly2 structures, resulting in a rigid connection. Figure 6.5(a) is an SEM of the multiplier. Figure 6.5(b) is an SEM showing a closeup of this error.

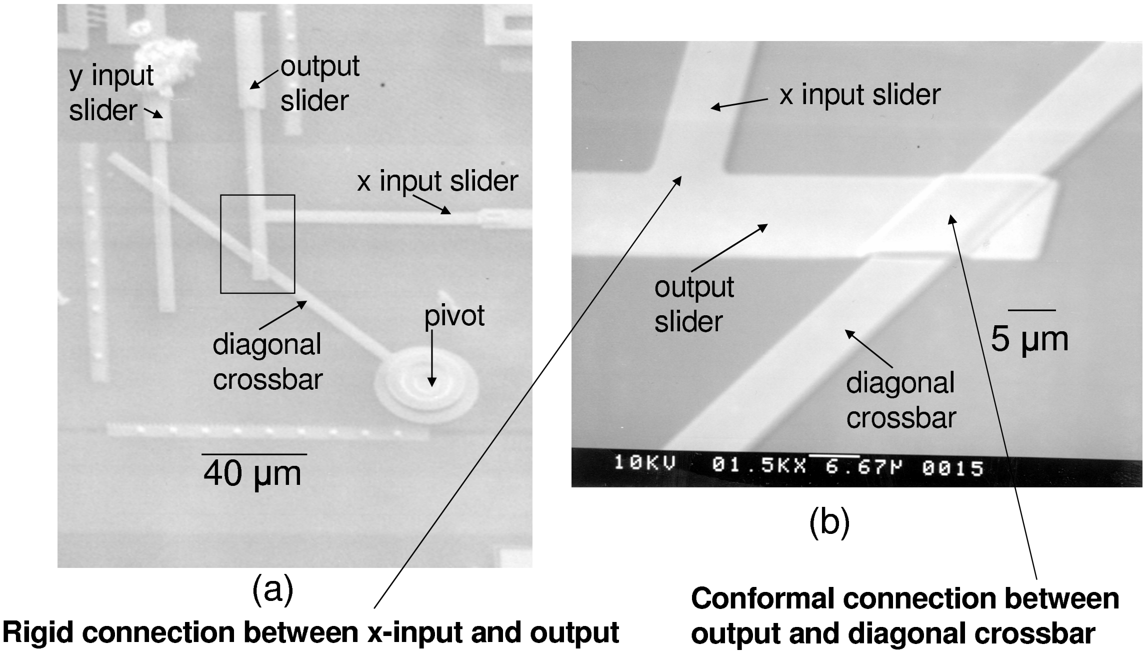


Figure 6.5 SEM of the multiplier showing the improper, rigid connection of the x-input to the output. (a) SEM of the multiplier. (b) SEM showing a closeup of the improper connections. Proper connections are made using the conformal process, as shown by the output connected to the multiplier crossbar.

As a result, the output bar, as shown in Figure 6.5, is unable to move in the vertical direction, since the x-input is prohibited from sliding vertically by the poly2 guide, and the x-input and output are rigidly connected. The new design addresses this problem by connecting the input and output appropriately, allowing the poly2 x-input to overlap and conform to the poly1 output bar, just as the poly2 portion of the output bar conforms to the multiplier crossbar in Figure 6.5. Figure 6.6 shows a computer aided drawing of the new multiplier design that should operate as expected. One potential problem that may occur is slipping of the poly2 conformal connections, since they are not secured in place with any kind of rigid support. If this were to occur, the pieces would need to be reconnected before operating the device.

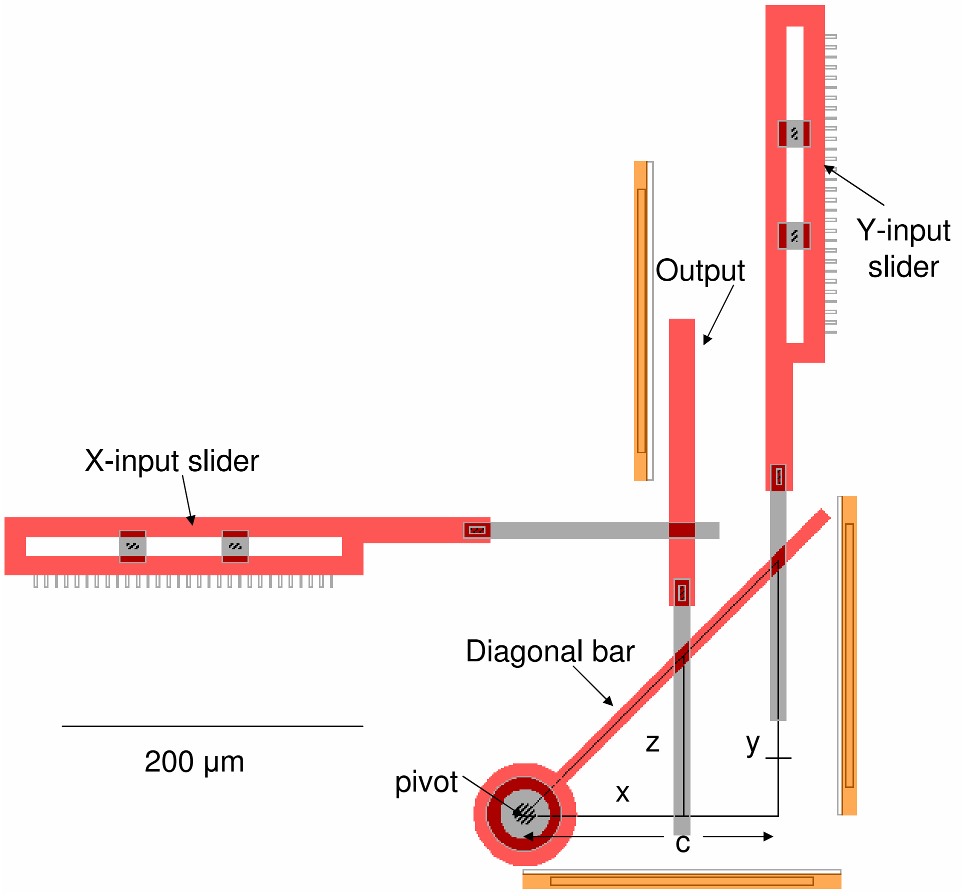


Figure 6.6 Computer aided drawing of the new multiplier design. This design takes advantage of the conformal layers of polysilicon, with the poly2 x-input overlapping the poly1 portion of the output slider. The poly2 y-input and the poly2 portion of the output slider overlap the poly1 diagonal crossbar for the same purpose, to allow for a semi-confined free moving structure. The x and y input sliders are confined to horizontal and vertical movements, respectively, by the poly2 guides, but the output slider is free to move in both directions, as guided by the inputs.

6.4 Integrators

The integrator does not work, since nothing was able to prevent the poly1-poly2 stacked pin for the poly2 gear from washing away during the release process, as shown in Figure 6.7(a). In some cases, this allowed the entire poly2 rotor to come unattached and flip over, as in Figure 6.7(b). Although the integrator can be improved upon by removing the top poly2 gear, this device will not work unless a third polysilicon layer is used to ensure that the poly1-poly2 pin does not wash away. A three-layer process is needed to realize this device. One improvement that can be made, however, regardless of what process is used to fabricate the device in the future, is that the output gear can be eliminated and the output slider can be connected directly to the friction wheel without affecting proper device operation. One possible way of securing the poly2 friction wheel without sacrificing its ability to move laterally is by connecting a small poly1 hub to a poly3 hub by way of a poly1-poly3 via. As mentioned previously, this device can not be realized with only

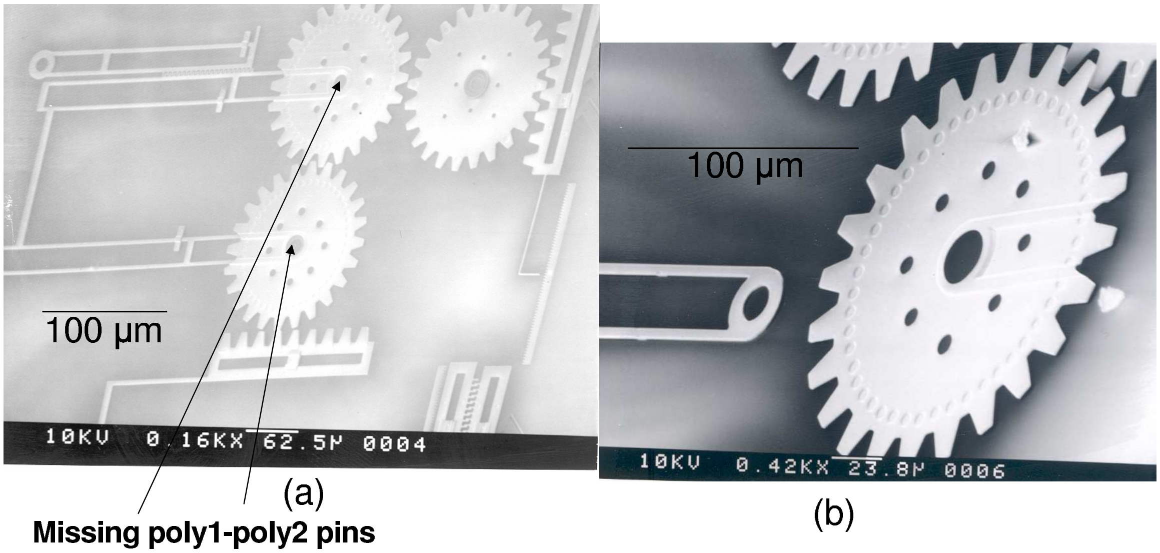


Figure 6.7 SEM of the integrator. (a) The poly1-poly2 stacked pins washed away during the release process. (b) The poly2 integrator gear flipped over during the release process.

two releasable layers. The poly3 layer, which is not available in the PolyMUMPs process, would serve as the third releasable layer.

6.5 Inverters

The inverter works as expected. Switching the input between states causes the gear to rotate and switch the output to the opposite state. Figure 6.8 is a video image that was captured during testing of the inverter.

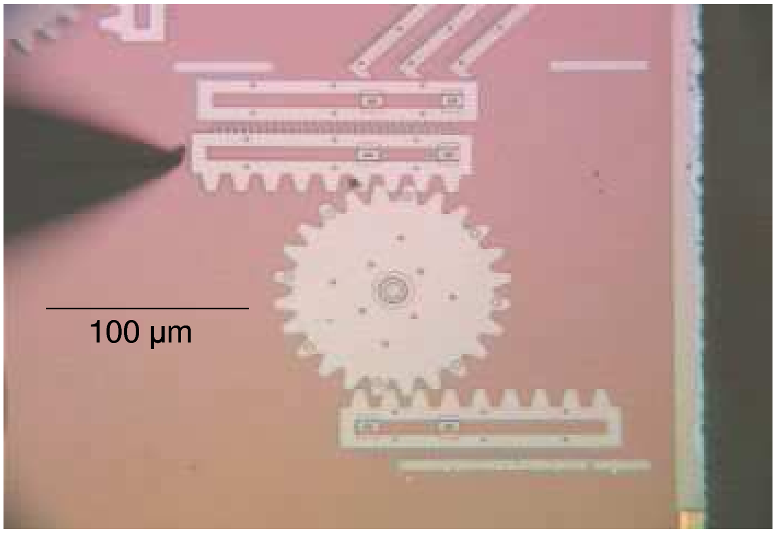


Figure 6.8 Picture of the inverter under testing. The input to the inverter has been switched and is about to be switched back.

6.6 Digital-to-analog converters

For the most part, the digital-to-analog converter, as shown in Figure 6.9, operated as expected. One design problem that was encountered and circumvented to allow for device testing was the position of the ruler at a considerable distance from the true output. The ruler was pointed to by a small poly1 bar that, during the design stage, was expected to remain perpendicular to the ruler. However, as the input states were changed, the angle of the true output marker relative to its initial position varied slightly. This slight variation was magnified greatly at the ruler-end

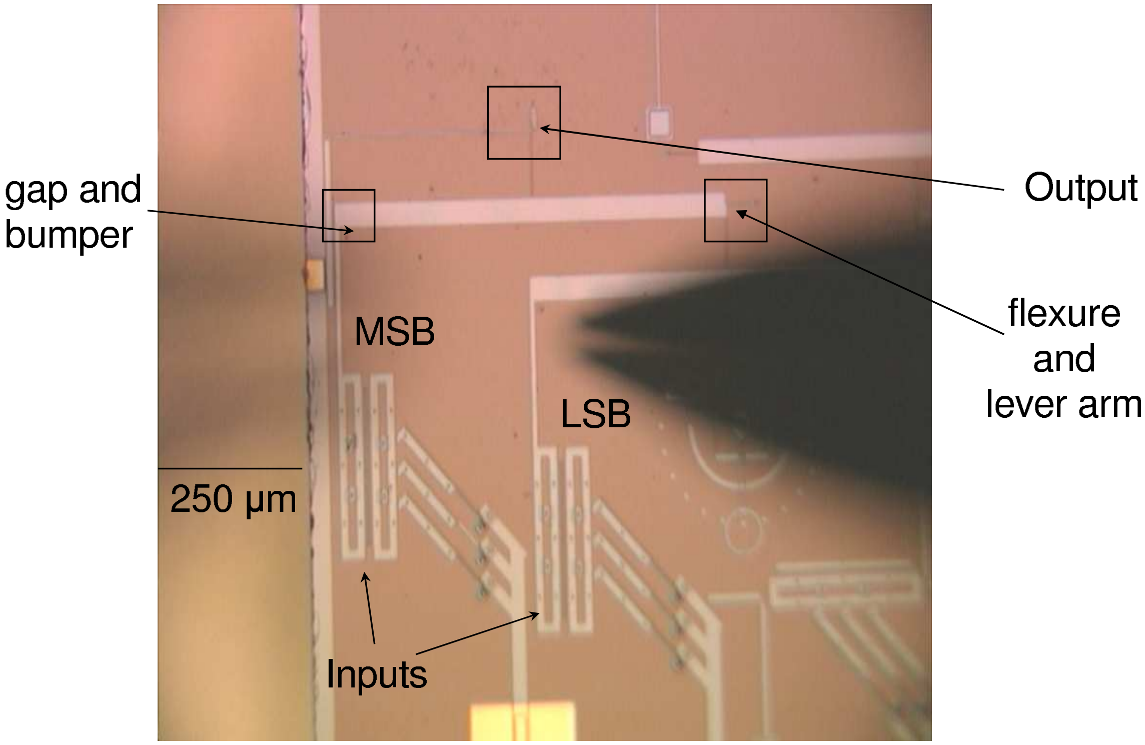
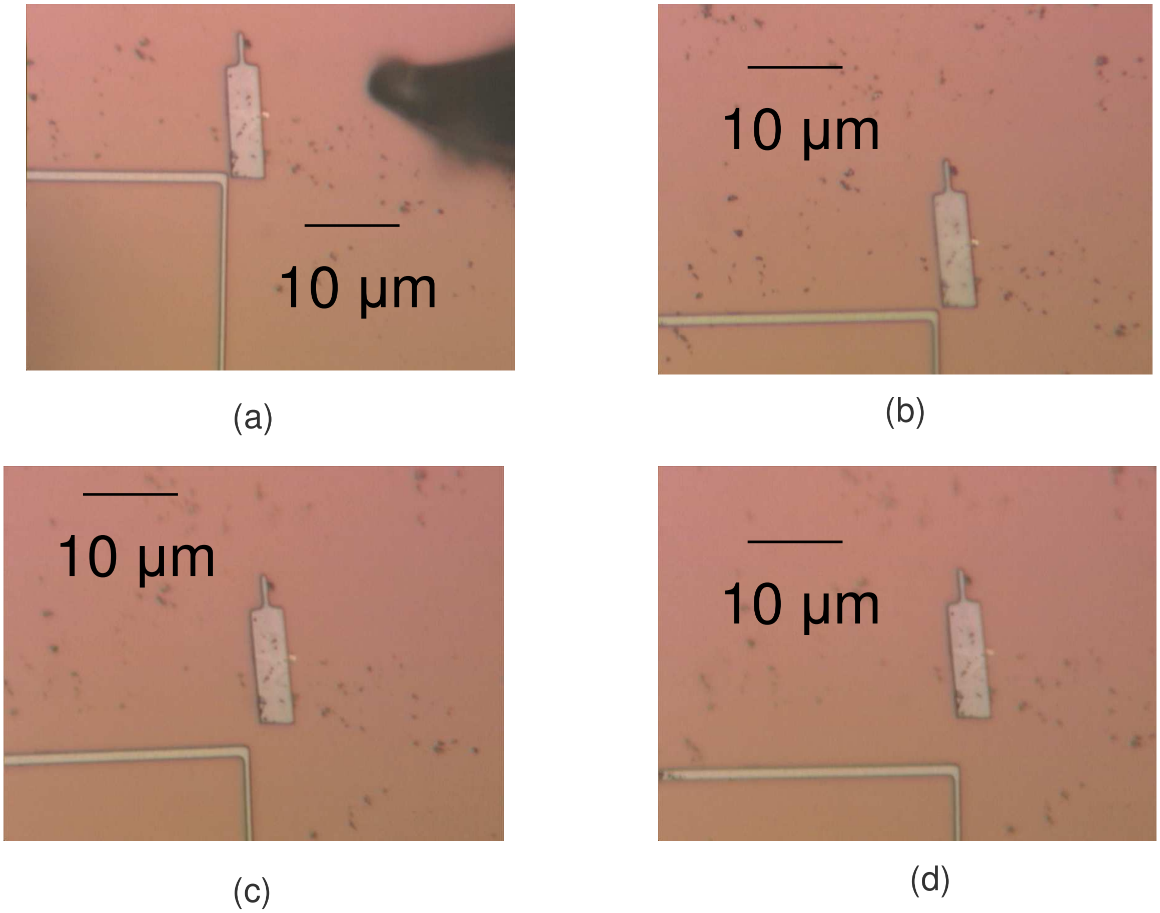


Figure 6.9 Video image of entire 2-bit D-to-A converter with specific regions labelled for future discussion.

of the bar, due to the length of the pointer that was fabricated perpendicular to the true output reference. For testing purposes, it was necessary to measure the output from the true output position, instead. This was done by making measurements relative to a structure that was positioned next to it. Figure 6.10(a) shows a video image of the output to the D-to-A converter prior to testing with a piece of poly1 from another device placed at the output as a marker for measurement purposes. With reference to Figure 6.9, the region that is being viewed in Figure 6.10 is labelled "Output". Figure 6.10(a) shows the output with both inputs set to 0. Figure 6.10(b) shows the output with the LSB set to 1 and the MSB set to 0. Figure 6.10 (c) shows the output with the MSB set to 1 and the LSB set to 0. Figure 6.10(d) shows the output with both inputs set to 1. Once data values were obtained for all input combinations, these were plotted against those expected based on the function derived for the D-to-A converter in Equation 3.14. The drastic differences between



(a)

(b)

(c)

(d)

Figure 6.10 Output of a 2-bit D-to-A converter in operation with a poly1 marker that has been positioned at the output for measurement purposes. (a) 00 input. (b) 01 input. (c) 10 input. (d) 11 input.

the experimental and predicted values prompted further investigation, especially since the experimental data appeared to be operating properly at the time of testing. During the investigation, two key issues were discovered.

First, an analysis of the video images that were captured during testing showed that the movement of the lever arm was not stopped by the bumper, which was located a distance of $g = 6$ μm from the original position of the lever arm, but rather by the anchor that connected the bumper to the substrate, which was a distance of $g_{real} = 9$ μm away. Figure 6.11 shows an SEM of the bumper and lever arm. As a reference, this picture corresponds to the region labelled “gap and bumper” in

Figure 6.9. The lever arm was able to slide underneath the bumper three additional microns, as shown in Figure 6.12.

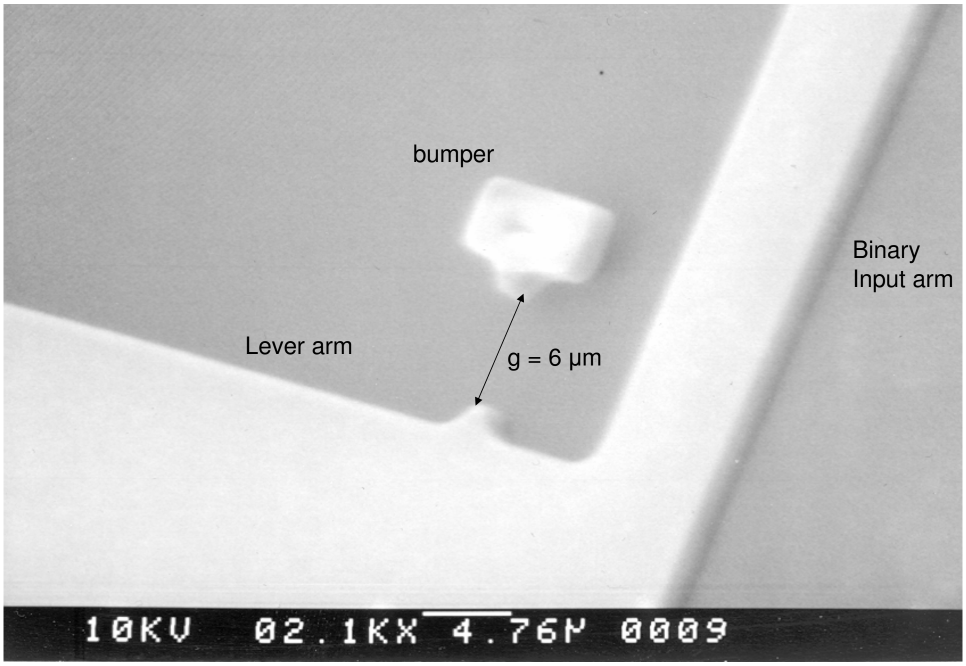


Figure 6.11 SEM of the bumper and lever arm of a D-to-A converter positioned with a gap of 6 μm.

Second, the output for each stage was not fabricated in the center of the lever arm, but at a distance $\alpha = \frac{406}{755}$ of the distance from the bumper to the end of the lever arm. Based on this, a new equation for output of the D-to-A converter was derived as

$$\begin{aligned} Output &= g\alpha B_{N-1} + \beta Output_{N-1} \\ &= \alpha g \sum_{i=0}^{N-1} \beta^{N-(i+1)} B_i \\ &= \alpha g \sum_{i=0}^{N-1} \frac{\beta^{N-1}}{\beta_i} B_i \qquad (6.1) \end{aligned}$$

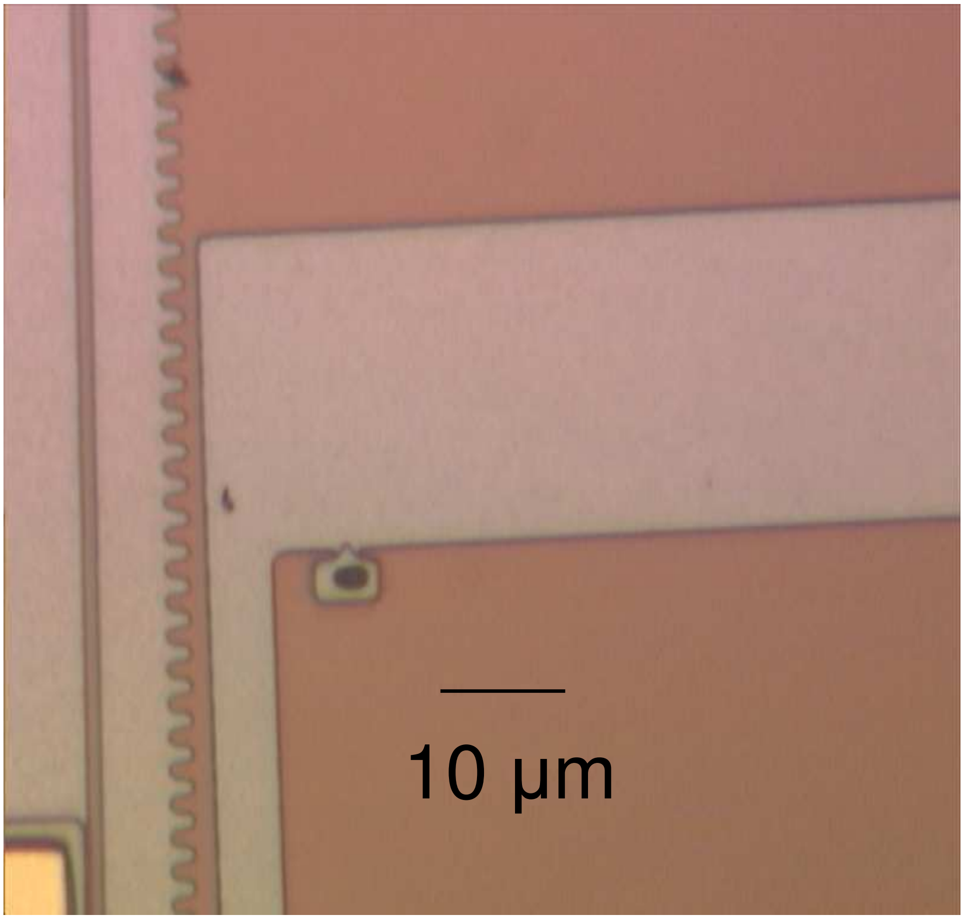


Figure 6.12 Lever arm travels three additional microns by sliding under the poly1 bumper until it is stopped by the anchor.

where β = 1 - α, and the other values are the same as those defined for Equation 3.14.

Figure 6.13 shows a plot of the original function from Equation 3.14, the experimental data values, and the new function from Equation 6.1 for a 2-bit D-to-A converter. The data values follow very closely to the new function with the values beginning to spread with both inputs are high. This is due to the final design issue that was discovered during device testing.

Due to inconsistent bending of the flexures connected to the pivot points, the output was not purely linear, especially when a lever arm was being "pulled" on both

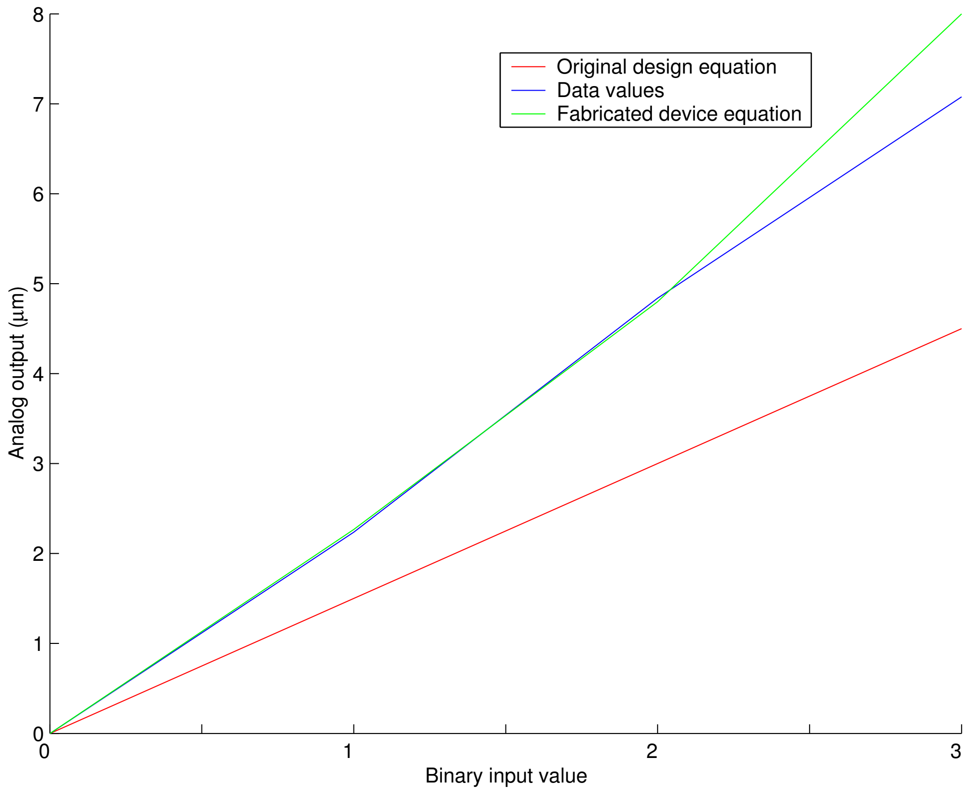


Figure 6.13 Plot of the 2-bit D-to-A converter data and equations.

sides by the input and output of the lower level. There was some variation due to bending of the flexure, which caused the pivot point of the lever arm, in some cases, to rotate slightly up, rather than to stay fixed. This excessive bending of the short flexure often led to device failure, as well, since the bent flexure was easily broken when all inputs were applied for a D-to-A convert with more than two bits. Longer flexures will more than likely remedy this problem for future designs. Figure 6.14 shows an SEM of the flexure connected to the lever arm. With reference to Figure 6.9, the SEM shows the "flexure and lever arm" region.

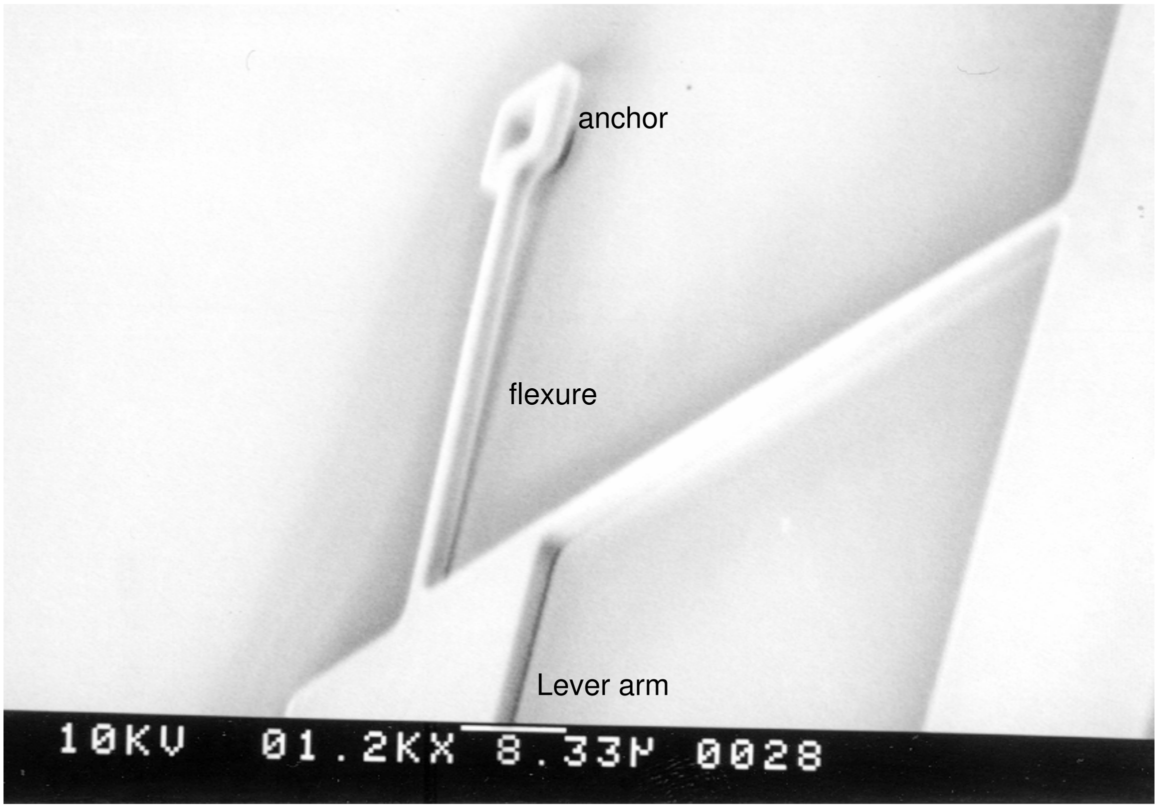


Figure 6.14 SEM of the flexure connected to a lever arm of a D-to-A converter. The flexure is 50 μm long.

This concludes the analysis of the designs that were tested for this research. Improvements have been made since these devices were fabricated, and these new and improved designs will be fabricated and tested as future work.

VII. Conclusions

This chapter serves as the conclusion to this thesis. It summarizes the work that was completed, discusses challenges that were encountered along the way, mentions lessons that were learned in the process, lists contributions that were made to the advancement of MEMS research, and recommends future work based on the results from this research.

7.1 Summary

This thesis began by discussing the motivations for purely mechanical computing devices in MEMS. These motivations include harsh environment operation of computing devices, such as high temperature and radiation environments, as well as the potential to develop smaller computing systems that contain passive mechanical devices.

Second, the history of mechanical computing devices was discussed, along with a discussion of topics related to the motivation for this research. This was followed by a discussion of MEMS, including the three main fabrication processes of surface micromachining, bulk micromachining, and micromolding. This was followed by a detailed explanation of the surface micromachining process that was used for the devices in this thesis. A detailed background of mechanical computing devices in MEMS was presented in order to effectively show the place that this research has in the context of related work in the field.

Third, the mechanical computing device designs were presented. This included a description of the operating principles of the device, including a derivation of the resulting equation or function. Reference was also made to the inspirations for the designs, many of which were of historical macro-scale mechanical devices, of which attempts were made to fabricate them as MEMS devices.

Fourth, a model of force and switching speed was derived for the digital computing devices. Each device was broken down into small device elements, and each element was modelled separately. Following the elemental modelling, an analysis of the speed and force requirements of the digital computing devices was undertaken using these device element models as a basis set.

Fifth, the experimental procedures that were followed for this research were discussed. The procedure section included the MEMS release process, as well as a discussion of equipment used and the methods employed for data acquisition in the form of video, video images, and SEMs.

Sixth, the results were presented, which discussed design problems that were encountered for devices that did not work as expected. These errors were analyzed, and corrections were made in the form of modified designs that will be fabricated in the next scheduled fabrication run.

This chapter serves as the final step in the research process. Experimental challenges, lessons learned, contributions made, and future work are presented. The next section discusses the experimental challenges that were encountered.

7.2 Experimental Challenges

Several challenges were encountered during the research process. Some challenges were self-inflicted, and others were a consequence of uncontrollable circumstances. One of the biggest challenges that had to be overcome was the lack of experience in creating mechanical designs. This challenge is constantly becoming less of a factor, as experience is gained by repeatedly designing MEMS devices and learning from design successes and failures. A second challenge was attempting to align the progress of the research with the scheduled commercial PolyMUMPs fabrication runs. With approximately a two month turnaround between design submission and device fabrication, and a month and a half time period between design submission dates, mistakes that are made on one design run tend to be repeated on the subsequent design run, since design errors are usually discovered during testing, which does not occur until after the second set of designs has already been submitted.

7.3 Lessons Learned

One of the most important lessons that was learned through this research process was the importance of the design stage. It is during this stage of research, for a project such as this, that all is won or lost. Devices that do not operate as expected must be redesigned, which results in a delay in the research process that may be avoided by designing the devices correctly the first time.

7.4 Contributions Made

This research resulted in the successful design of a purely mechanical inverter, sine and cosine functions, and a digital-to-analog converter. It is expected that the other designs that were presented in Chapter III will also result in successful devices. Among the devices that are to be fabricated for future testing are the differential, NAND, NOR, and XOR logic gates, and an analog-to-digital converter. The improved tangent/cotangent function and multiplier, which were discussed in Chapter VI will be fabricated for future testing as well. Assuming these devices operate as designed, this research will make a significant contribution to work in the area of mechanical computing in MEMS.

7.5 Future Work

This thesis has taken a small step towards the fulfillment of the goal of MEMS purely mechanical computing devices that can operate in harsh environments. More work is required in order to achieve this goal. This section lists some of the future work that is expected to advance this research further.

7.5.1 Testing of new and improved mechanical computing devices. The first step in the process of advancing this research is testing the new devices that have been designed and are in the process of being fabricated. These devices should be released and tested to determine if they operate as expected. If not, new designs should be made to correct any errors that are found.

7.5.2 Experimentally verify the mechanical models. The mechanical models for digital devices have been derived, but they have not been verified experimentally. The first step to verifying the models, once the devices have been fabricated and released, is to experimentally determine the true damping and friction coefficients, as well as the resulting damping and friction forces and torques for each element. As with the mechanical models that were derived in Chapter IV, once each element has been analyzed and the models have been either verified or corrected, changes can be made to the models, and the new device models can be experimentally verified.

7.5.3 Device optimization. All devices, whether they work or not, can be optimized in one way or another. Optimization could be done to make the devices as small, fast, or durable as possible. Based on the equations that were derived for modelling the device switching speeds, decreasing the size of the devices by one order of magnitude increases the switching speed 10 times. Decreasing size by two orders of magnitude results in nearly a 1000 time increase in switching speed. A three order decrease in size results in a 10000 time increase in switching speed, and so forth. In other words, size reduction and fast switching speeds can be addressed simultaneously in the optimization process. Once the desired optimization objective is met, the optimized device should be fabricated and tested for correct operation.

7.5.4 Mechanical computing designs using other fabrication processes. As was discussed with regards to some of the devices for this research, including integrators and trigonometric functions, another fabrication process that has more structural layers that are planarized, rather than conformal would be advantageous. For example, the SUMMIT V process consists of two planarized structural polysilicon layers. Designing devices in more flexible fabrication processes may provide the necessary conditions to either realize a function that is not possible in the more limited MUMPs® process, or may allow for further optimization that can not be realized with the MUMPs® process.

7.5.5 Harsh environment operation of mechanical computing devices.

Once optimized devices have been fabricated in a multitude of fabrication processes, the devices should be tested in harsh environments, such as environments with high radiation and high temperature, to determine if the mechanical computing devices serves the purpose of their creation. Along those lines, a method should be developed to allow for actuation of the device while in the harsh environment without requiring the operator to be exposed to the harsh environment.

7.6 Conclusion

This concludes the thesis research. Many different purely mechanical computing devices in MEMS have been designed. Some have been fabricated and tested. The rest will be fabricated and tested in the future. Much work has been done, but much work remains left undone. This specific area of MEMS is ripe for research. The suggestions that have been presented convey my personal thoughts as to how future work should be directed, but one might think of other, possibly better, methods to move this work forward. Regardless, my hope is that this is not the end to this research, but rather a launching point from which others might be able to pick up where this work has left off.

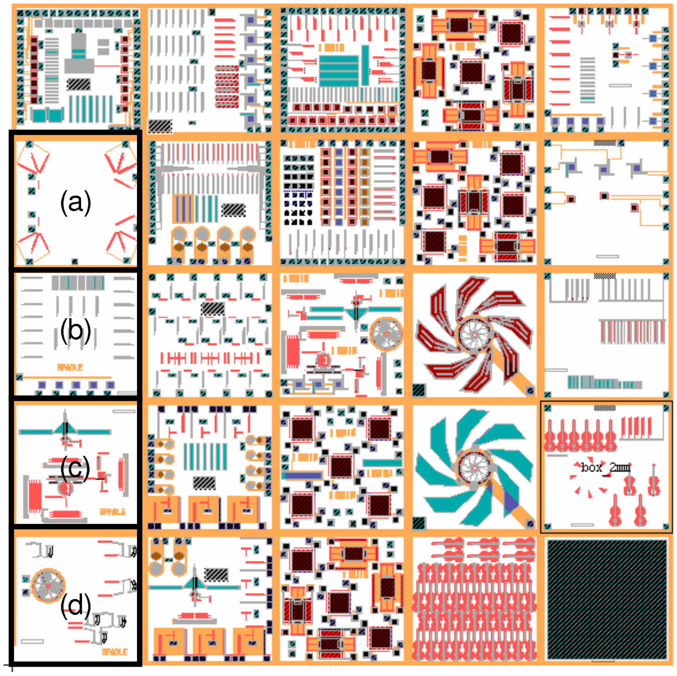


Figure A.1 Computer aided drawing of the MUMPs 48 design layout. (a) Electrostatic-Etherm NAND gates. (b) Electrothermal actuators, Electrostatic piston mirrors, and Residual stress cantilevers. (c) Pop-up F-16 and Scratch drive. (d) Scratch drive and Kladitis' logic gates.

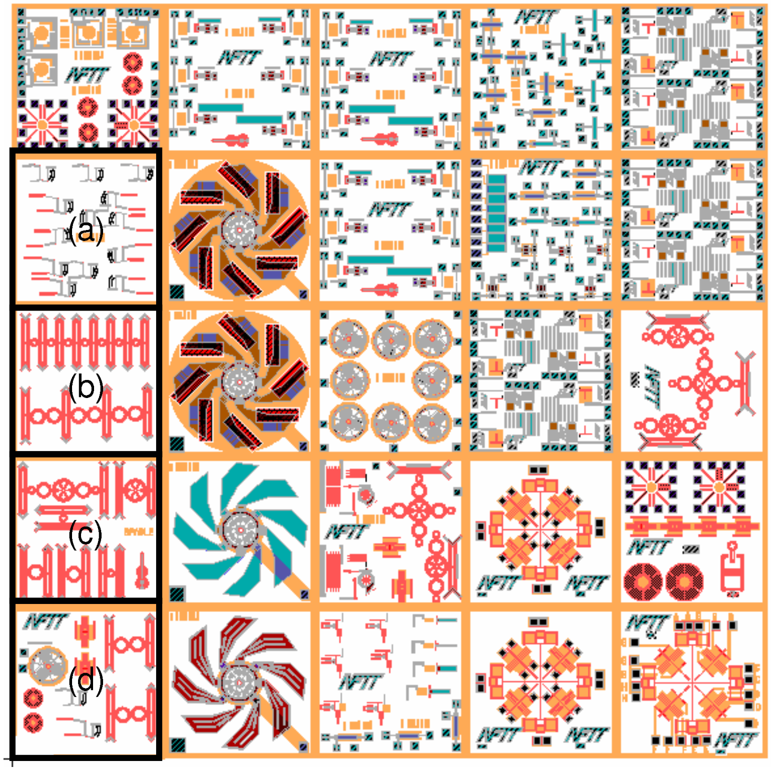


Figure A.2 Computer aided drawing of the MUMPs 49 design layout. (a) Kladitis' logic gates. (b) Gear arrays. (c) Inverters and gear arrays. (d) Scratch drive, Kladitis' logic gates, and Buffers.

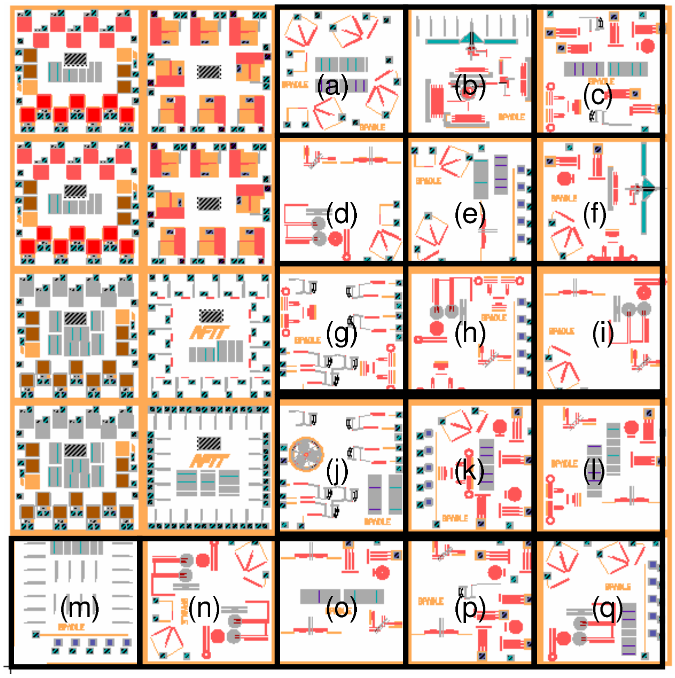


Figure A.3 Computer aided drawing of the MUMPs 51 design layout. (a) Electrostatic-Etherm logic gates and residual stress cantilevers. (b) Pop-up F-16 and micromotor. (c) Residual stress cantilevers, Electrothermal actuator arrays, Differential (adder), and inverter. (d) Integrator, Electrostatic-Etherm logic gate, and sine/cosine function device. (e) Residual stress cantilevers, Electrostatic piston mirrors, Electrostatic-Etherm logic gates, and tangent/cotangent function device. (f) Pop-up F-16, Inverter, Electrostatic-Etherm logic gate, Differential (adder), and Electrothermal actuator array. (g) Differential (adder) and Kladitis' logic gates. (h) Integrator, Differential (adder), Electrostatic piston mirrors, and sine/cosine function device. (i) Integrator, sine/cosine function device, Electrostatic-Etherm logic gate, and tangent/cotangent function device. (j) Kladitis' logic gates, residual stress cantilevers, and scratch drive. (k) Residual stress cantilevers, Electrostatic piston mirrors, Electrostatic-Etherm logic gates, Electrothermal actuator arrays, and inverter. (l) Residual stress cantilevers, Differential (adder), tangent/cotangent function device, Electrothermal actuator arrays, sine/cosine function device, and inverter. (m) Residual stress cantilevers, Electrostatic piston mirrors, and Electrothermal actuators. (n) Integrator and Electrostatic-Etherm logic gates. (o) Residual stress cantilevers, tangent/cotangent function device, sine/cosine function device, electrothermal actuator arrays, and inverter. (p) Sine/cosine function device, Electrothermal actuator arrays, Kladitis' logic gates, and inverter. (q) Electrostatic-Etherm logic gates, integrator, residual stress cantilevers, and electrostatic piston mirrors.

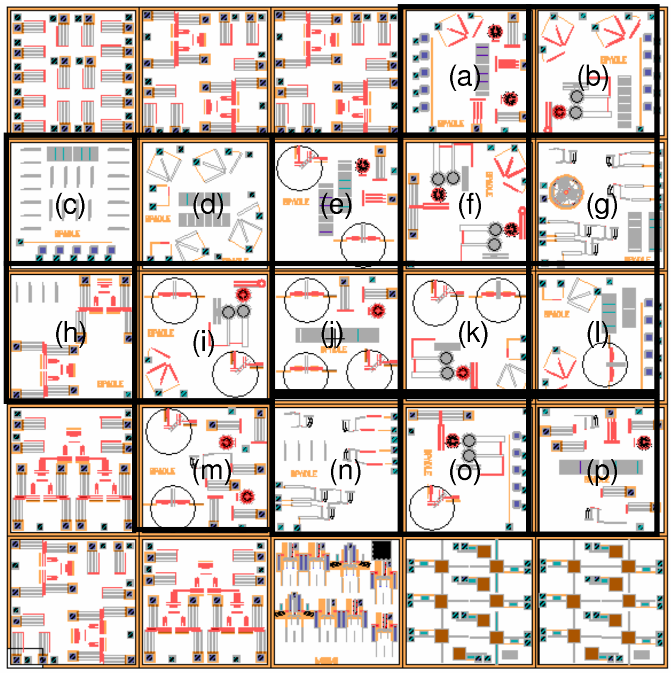


Figure A.4 Computer aided drawing of the MUMPs 52 design layout. (a) Electrothermal actuator arrays, electrostatic piston mirrors, residual stress cantilevers, Electrostatic-Etherm logic gates, and inverter. (b) Integrator, residual stress cantilevers, electrostatic piston mirrors, and electrostatic-etherm logic gates. (c) Electrothermal actuators, residual stress cantilevers, and electrostatic piston mirrors. (d) Residual stress cantilevers and electrostatic-etherm logic gates. (e) Residual stress cantilevers, sine/cosine function device, tangent/cotangent function device, inverter, and electrothermal actuator arrays. (f) Integrators and electrostatic-etherm logic gates. (g) Scratch drive, Kladitis' logic gates, and residual stress cantilevers. (h) Differential (adder) and electrothermal actuators. (i) Integrator, sine/cosine function device, tangent/cotangent function device, and electrostatic-etherm logic gates. (j) Residual stress cantilevers, tangent/cotangent function device, sine/cosine function device, electrothermal actuator array, and inverter. (k) Integrator, sine/cosine function device, tangent/cotangent function device, and electrostatic-etherm logic gate. (l) Residual stress cantilevers, electrostatic piston mirrors, electrostatic-etherm logic gates, and tangent/cotangent function device. (m) Sine/cosine function devices, tangent/cotangent function devices, inverters, and electrothermal actuator arrays. (n) Kladitis' logic gates and electrothermal actuators. (o) Integrator, electrostatic piston mirrors, and sine/cosine function device. (p) Residual stress cantilevers, electrothermal actuator arrays, and inverter.

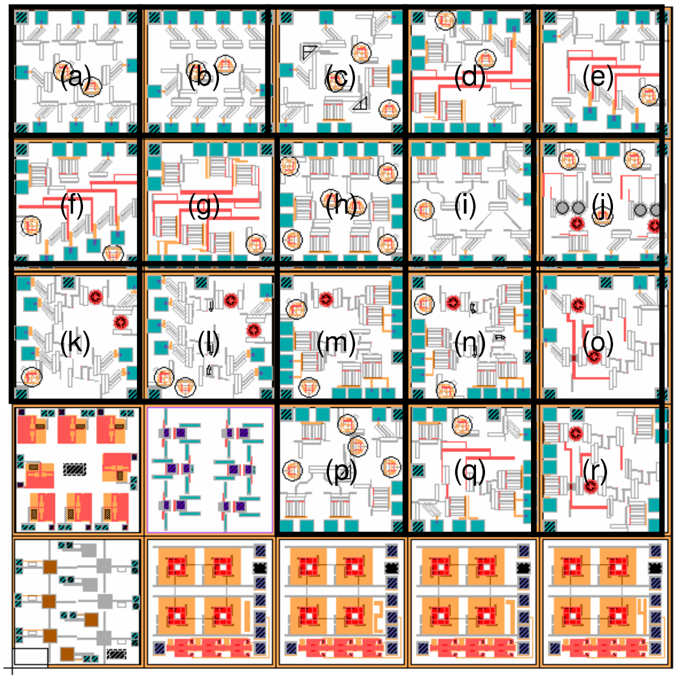


Figure A.5 Computer aided drawing of the MUMPs 53 design layout. (a) Vibromotor force test structures, sine/cosine function devices, and tangent/cotangent function devices. (b) Vibromotor force test structures, since/cosine function devices, and tangent/cotangent function devices. (c) Multiplier, vibromotor force test structures, sine/cosine function devices, and tangent/cotangent function devices. (d) 2-bit D-to-A converters, electrothermal actuator force test structures, vibromotor force test structures, since/cosine function devices, and tangent/cotangent function devices. (e) 3-bit D-to-A converter, Vibromotor force test structures, and sine/cosine function device. (f) 4-bit D-to-A converter, electrothermal actuator force test structures, vibromotor force test structures, sine/cosine function devices, and tangent/cotangent function devices. (g) 2-bit and 3-bit D-to-A converters, electrothermal actuator force test structures, and sine/cosine function devices. (h) Electrothermal actuator force test structures, sine/cosine function devices, and tangent/cotangent function devices. (i) Differentials (adders), vibromotor force test structures, and tangent/cotangent function devices. (j) Integrator, electrothermal actuator array, electrothermal actuator force test structure, vibromotor force test structure, sine/cosine function device, and tangent/contangent function device. (k) Inverters, sine/cosine function device, and differentials (adders). (l) Inverters, differentials (adders), sine/cosine function devices, and tangent/cotangent function devices. (m) Sine/cosine function device, tangent/cotangent function device, and an AND gate. (n) AND gate, differential (adder), sine/cosine function device, and tangent/cotangent function device. (o) XOR gate and vibromotor test structure. (p) Differential (adder), sine/cosine function devices, tangent/cotangent function devices, vibromotor force test structures, and electrothermal actuator force test structures. (q) 2-bit D-to-A converter, vibromotor force test structures, differential (adder), and sine/cosine function device. (r) XOR gate, vibromotor force test structure, and electrothermal actuator force test structure.

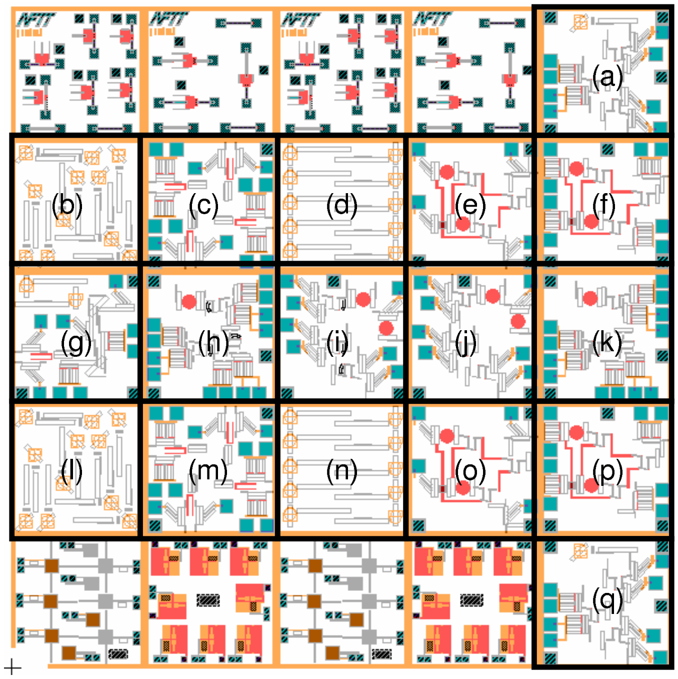


Figure A.6 Computer aided drawing of the MUMPs 54 design layout. (a) Differentials (adders) and cosine function device. (b) Sine function device and cosine function device. (c) Vibromotor force test structures and electrothermal force test structures. (d) Tangent/cotangent function devices. (e) XOR gate and vibromotor force test structures. (f) XOR gate and electrothermal actuator force test structures. (g) Cosine function devices, Sine function devices, vibromotor force test structures, and multipliers. (h) AND gate and NAND gates. (i) AND gates, inverters, and vibromotor force test structures. (j) AND gates, inverters, and vibromotor force test structures. (k) Electrothermal actuator force test structures and an AND gate. (l) Sine function devices and cosine function devices. (m) Vibromotor force test structures and electrothermal actuator force test structures. (n) Tangent/cotangent function devices. (o) XOR gate and vibromotor force test structures. (p) XOR gate and electrothermal actuator force test structures. (q) Cosine function device, electrothermal actuator force test structures, and vibromotor force test structures.

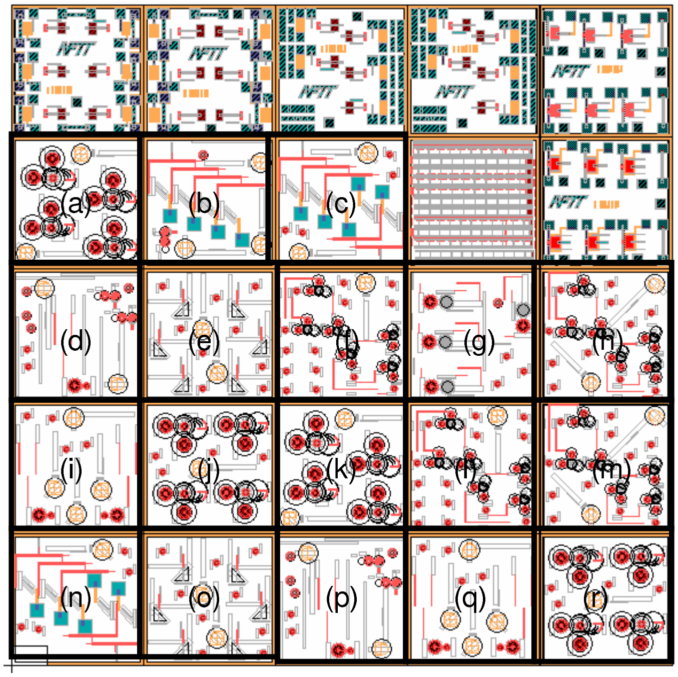


Figure A.7 Computer aided drawing of the MUMPs 55 design layout. (a) NAND logic gates and cosine function devices. (b) 4-bit D-to-A converter, sine function devices, cosine function devices, and mechanical A-to-D converter. (c) 2-bit and 3-bit D-to-A converters, inverters, sine function device, and cosine function device. (d) Differential (adder), cosine function devices, mechanical A-to-D converters, and inverters. (e) Multipliers, inverters, sine function devices and cosine function devices. (f) XOR logic gates, inverters, and sine function device. (g) Integrators and inverters. (h) XOR gats, sine function device, cosine function device, and inverters. (i) Sine function devices, cosine function devices, and differentials (adders). (j) NOR logic gates, inverters, and sine function devices. (k) NAND logic gates and cosine function devices. (l) Inverers, XOR logic gates, and sine function device. (m) XOR logic gate tangent/cotangent function devices, and inverters. (n) 2-bit and 3-bit D-to-A converters, cosine function device, and inverters. (o) Multipliers, inverters and sine function devices. (p) Mechanical A-to-D converters, inverters, and tangent/cotangent function devices. (q) Tangent/cotangent function devices, inverters, and differentials (adders). (r) NOR logic gates, inverters, and cosine function devices.

Bibliography

1. Wondrak, W., “Physical Limits and Lifetime Limitations of Semiconductor Devices at High Temperatures,” *Microelectronics Reliability*, vol. 39, no. 6, pp. 1113–1120, 1999.

2. Kladitis, P.E., “MEMS Micro-Mechanical Logic Gates for Mechanical Computing in Machine Only Environments,” *Solid-State Sensor, Actuator, and Microsystems Workshop: Hilton Head Island, South Carolina*, 2002.

3. Toshiyoshi, H., D. Kobayachi, M. Mita, G. Hashiguchi, H. Fujita, J. Endo, and Y. Wada, “Micro Electro Mechanical Digital-to-Analog Converter (MEMDAC),” *10th International Conference on Solid-State Sensors and Actuators: Transducers ’99*, 1999.

4. Toshiyoshi, H., D. Kobayashi, M. Mita, G. Hashiguchi, H. Fujita, J. Endo, and Y. Wada, “A Digital-to-Analog Converter of Displacement by an Integrated Micromechanism,” *Japanese Journal of Applied Physics*, vol. 38, pp. 593–595, May 1999.

5. Toshiyoshi, H., D. Kobayashi, M. Mita, G. Hashiguchi, H. Fujita, J. Endo, and Y. Wada, “Microelectromechanical Digital-to-Analog Converters of Displacement for Step Motion Actuators,” *Journal of Microelectromechanical Systems*, vol. 9, pp. 218–225, June 2000.

6. Yeh, R., R. A. Conant, and K. S. Pister, “Mechanical Digital-to-Analog Converters,” *10th International Conference on Solid-State Sensors and Actuators: Transducers ’99*, 1999.

7. Hirata, A., K. Machida, H. Kyuragi, and M. Maeda, “A Micromechanical Switch as the Logic Elements for Circuits in Multi Chip Module on Si (MCM-Si),” *Proceedings of the 12th IEEE International Conference on MEMS*, 1999.

8. Hirata, A., K. Machida, H. Kyuragi, and M. Maeda, “A electrostatic micromechanical switch for logic operation in multichip modules on Si,” *Sensors and Actuators A: Physical*, vol. 80, pp. 119–125, 2000.

9. Kruglick, E. J. J. and K. S. J. Pister, “MEMS Relay Based Digital Logic Systems,” *10th International Conference on Solid-State Sensors and Actuators: Transducers ’99*, 1999.

10. Kruglick, E. J. J., *Microrelay design, performance, and systems.* PhD thesis, University of California, Berkeley, 1999.

11. “L-edit version 8.30.” Computer Software. Tanner Research, Inc. 2650 East Foothill Boulevard Pasadena, CA 91107, USA.

12. Koester, D. A., R. Mahadevan, B. Hardy, and K. W. Markus, "MUMPs® Design Handbook: Revision 7.0." JDS Uniphase, MEMS Business Unit, 2001.

13. Dorf, R. C., *Computers and Man.* San Francisco, CA: Boyd and Fraser Publishing Company, 1974.

14. Augarten, S., *Bit by Bit: An Illustrated History of Computers.* New York: Ticknor and Fields, 1984.

15. Evans, C., *The Making of the Micro: A History of the Computer.* Victor Gollancz Ltd., 1981.

16. Ketelaars, N., "Pascal's calculator," *AIMe Magazine*, vol. 2, pp. 3–5, December 2001.

17. Eames, C. and R. Eames, *A Computer Perspective.* Cambridge, Massachussetts: Harvard University Press, 1973.

18. Clymer, A. B., "The Mechanical Analog Computers of Hannibal Ford and William Newell," *IEEE Annals of the History of Computing*, vol. 15, no. 2, pp. 19–34, 1993.

19. Drexler, K. E., *Nanosystems: Molecular Machinery, Manufacturing, and Computation.* John Wiley and Sons, Inc., 1992.

20. Feynman, R., "Infinitesimal Machinery," *Journal of Microelectromechanical Systems*, vol. 2, pp. 4–14, March 1993.

21. Wong, P. C., K.-K. Wong, and H. Foote, "Organic data memory: Using the dna approach," *Communications of the ACM*, vol. 46, pp. 95–98, January 2003.

22. Brown, R. B. and K. Wu, *High Temperature Electronics*, ch. 4. High temperature operation of silicon MOS transistors, pp. 67–119. London: Chapman and Hall, 1997.

23. van Lint, V. A. J., T. M. Flanagan, R. E. Leadon, J. A. Naber, and V. C. Rogers, *Mechanisms of Radiation Effects in Electronic Materials*, vol. 1. John Wiley and Sons, 1980.

24. Larin, F., *Radiation Effects in Semiconductor Devices.* New York: John Wiley and Sons, Inc., 1968.

25. Helvajian, H. and S. Janson, *Microengineering Aerospace Systems*, ch. Microengineering Space Systems. The Aerospace Corporation, 1999.

26. W. R. Dawes, F. B. McLean, P. A. Robinson, Jr., and J. J. Silver, "Hardening Semiconductor Components Against Radiation and Temperature," tech. rep., Noyes Data Corporation, Park Ridge, NJ, 1989.

27. J. W. Howard and D. Hardage, "Spacecraft Environments Interactions: Space Radiation and Its Effects on Electronic Systems," tech. rep., NASA TP-1999-209373, 1999.

28. Judy, J. W., “Microelectromechanical systems (MEMS): fabrication, design and applications,” *Smart Materials and Structures*, vol. 10, pp. 1115–1134, 2001.

29. Watanabe, Y., T. Mitsui, T. Mineta, S. Kobayashi, N. Taniguchi, and K. Okada, “Five-axis motion sensor with electrostatic drive and capacitive detection fabricated by silicon bulk micromachining,” *Sensors and Actuators A: Physical*, vol. 97-98, pp. 109–115, April 2002.

30. Madou, M., *Fundamentals of Microfabrication*, ch. 6:LIGA, pp. 275–323. CRC Press, 1997.

31. Wallrabe, U., P. Bley, B. Krevet, W. Menz, and J. Mohr, “Design rules and test of electrostatic micromotors made by the LIGA process,” *Journal of Micromechanics and Microengineering*, vol. 4, no. 1, pp. 40–45, 1994.

32. Kruglick, E. J. J. and K. S. J. Pister, “Lateral MEMS Microcontact Considerations,” *Journal of Microelectromechanical Systems*, vol. 8, pp. 264–271, September 1999.

33. Chironis, N. P., *Mechanisms, Linkages, and Mechanical Controls.* New York: McGraw Hill, 1965.

34. Krygowski, T. W., M. S. Rodgers, J. J. Sniegowski, S. M. Miller, and J. Jakubczak, “A low-voltage rotary actuator fabricated using a five-level polysilicon surface micromachining technology,” *International Electron Devices Meeting*, pp. 697–700, 1999.

35. Freyenhagen, J. Personal Correspondence, January 2003.

36. Li, L., G. Brown, and D. Uttamchandani, “Study of scratch drive actuator force characteristics,” *Journal of Micromechanics and Microengineering*, vol. 12, pp. 736–741, 2002.

37. Madou, M., *Fudamentals of Microfabrication.* New York: CRC Press, 1997.

38. Veijola, T., “Compact damping models for lateral structures including gas rarefaction effects,” *Proceedings of MSM 2000, San Diego, CA*, pp. 162–165, March 2000.

39. Chang, K., S.-C. Lee, and S.-H. Li, “Squeeze film damping effect on a MEMS torsion mirror,” *Journal of Micromechanics and Microengineering*, vol. 12, pp. 556–561, June 2002.

40. Omar, M. P. and M. Mehregany, “Electric and fluid field analysis of side-drive micromotors,” *Journal of Microelectromechanical Systems*, vol. 1, pp. 130–140, September 1992.

41. Gieck, K. and R. Gieck, *Engineering Formulas.* New York: McGraw Hill, 7 ed., 1997.

42. Cho, Y., A. P. Pisano, and R. T. Howe, "Viscous damping model for laterally oscillating microstructures," *Journal of Microelectromechanical Systems*, vol. 3, pp. 81–87, June 1994.

43. Dhuler, V. R., M. Mehregany, and S. M. Phillips, "A comparative study of bearing designs and operational environments for harmonic side-drive micromotors," *IEEE Transactions on Electron Devices*, vol. 40, pp. 1985–1989, November 1993.

44. Mehregany, M. and Y.-C. Tai, "Surface micromachined mechanisms and micromotors," *Journal of Micromechanics and Microengineering*, vol. 1, pp. 73–85, 1991.

45. Dyck, C. W., J. H. Smith, S. L. Miller, E. M. Russick, and C. L. J. Adkins, "Supercritical carbon dioxide solvent extraction from surface-micromachined micromechanical structures," *SPIE Micromachining and Microfabrication*, vol. 2879, pp. 225–235, October 1996.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
25-03-2003	Master's Thesis	Aug 2002 - Mar 2003

4. TITLE AND SUBTITLE

MECHANICAL COMPUTING IN MICROELECTROMECHANICAL SYSTEMS (MEMS)

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Kenneth C. Bradley, 2d Lt, USAF

5d. PROJECT NUMBER

ENR # 2002-027

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

AFIT/ENG
BLDG 642
2950 HOBSON WAY
WRIGHT PATTERSON AFB OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT/GE/ENG/03-04

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Scott L. Roberson, Ph.D.
Air Force Research Laboratory Fuzes Branch (AFRL/MNMF)
2306 Perimeter Road
Eglin AFB, FL 32542-5910
850-882-2005 ext. 257

10. SPONSOR/MONITOR'S ACRONYM(S)

AFRL/MNMF

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for Public Release; Distribution Unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Mechanical computing devices in polysilicon-based microelectromechanical systems (MEMS) were designed with the goal of developing computing devices for harsh environments, such as those with high dose radiation and high temperatures, as well as devices that may be able to interface with molecular or biological computer systems. The devices that were designed include both analog and digital computing devices. The analog devices include integrators, differentials (summers), multipliers, and those that perform trigonometric functions. The digital devices that were designed are inverters, NAND, NOR, and XOR logic gates. Analog-to-digital (A-to-D) and digital-to-analog (D-to-A) converters were also designed. The designs were submitted to a commercial surface micromachining foundry to be fabricated. The completed MEMS devices were then released and tested to determine proper operation. Of the mechanical devices that have been fabricated and tested, a functioning inverter, sine function device, cosine function device, and digital-to-analog converter have been demonstrated.

15. SUBJECT TERMS

Microelectromechanical Systems, Mechanical components, Differentials (Mechanical), Integrators (Computers), Logic devices, Logic gates, NAND Gates, NOR Gates,

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Paul E. Kladitis, Capt, USAF (AFIT/ENG)
					19b. TELEPHONE NUMBER *(Include area code)*
U	U	U	UU	172	937-255-3636 ext. 4595 paul.kladitis@afit.edu